

FIG. 1

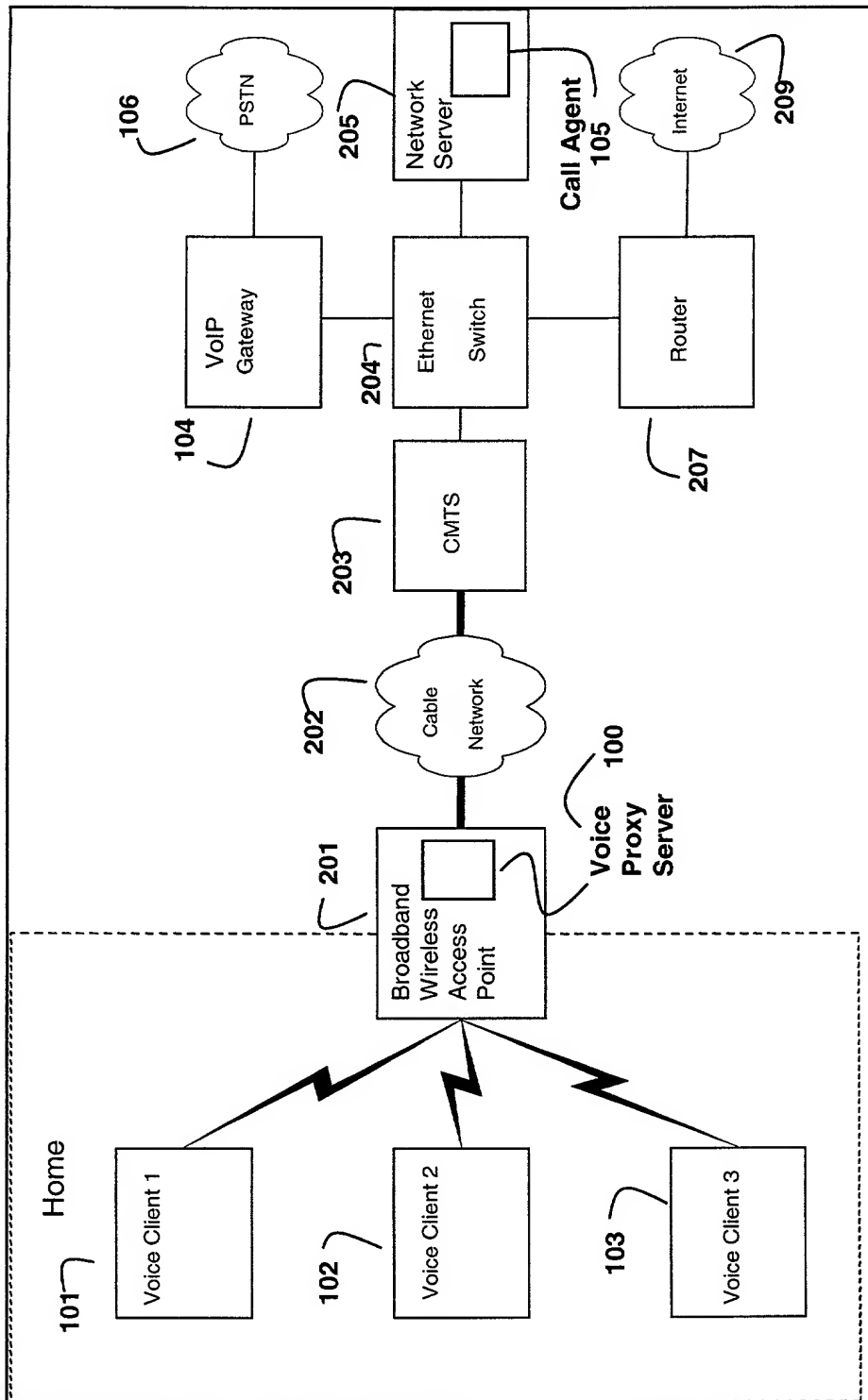
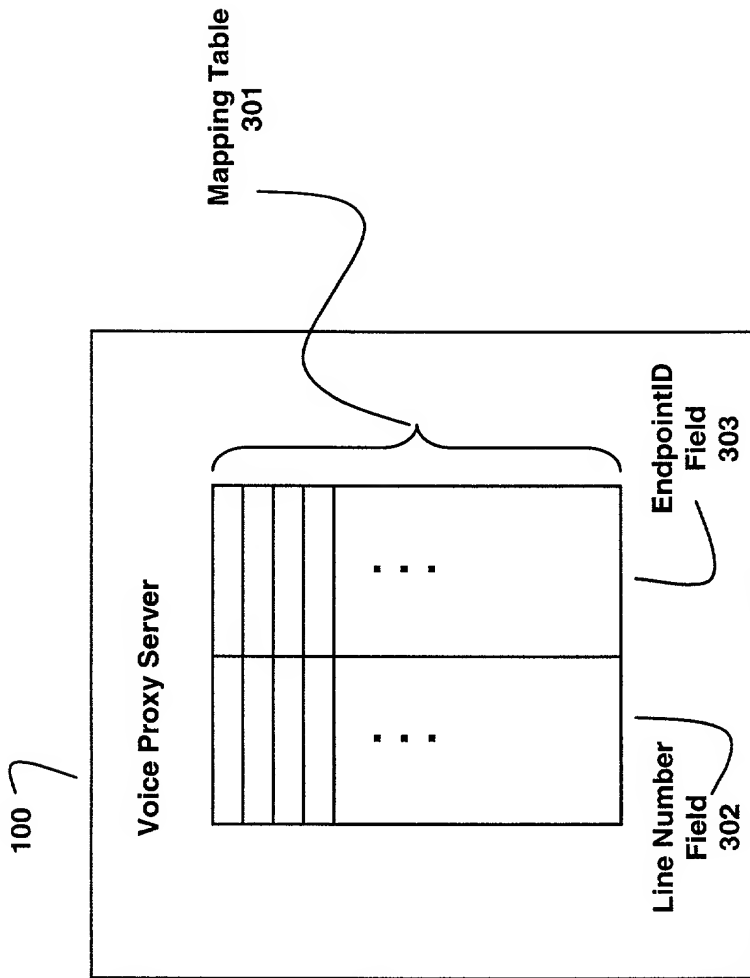
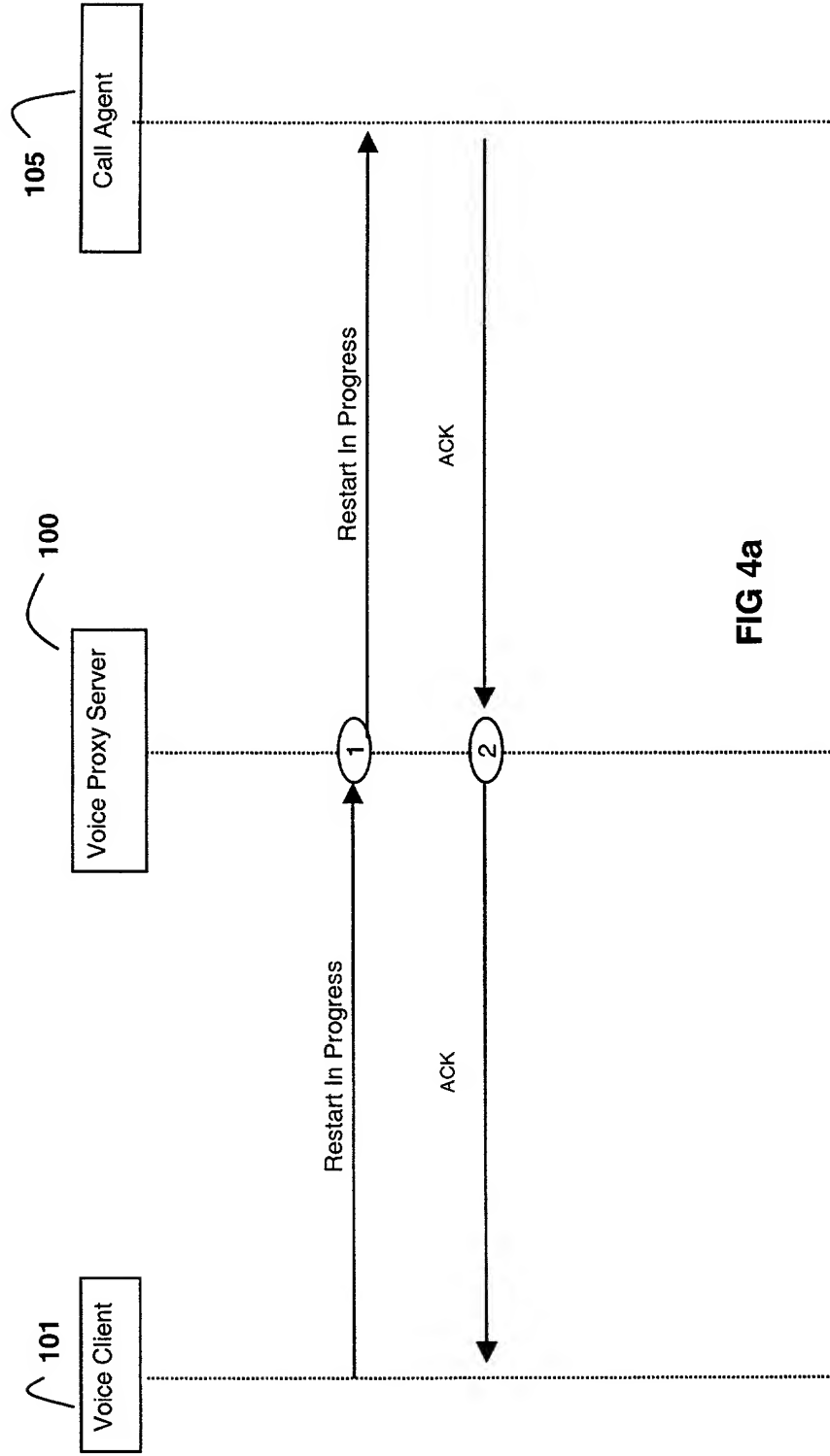


FIG. 2



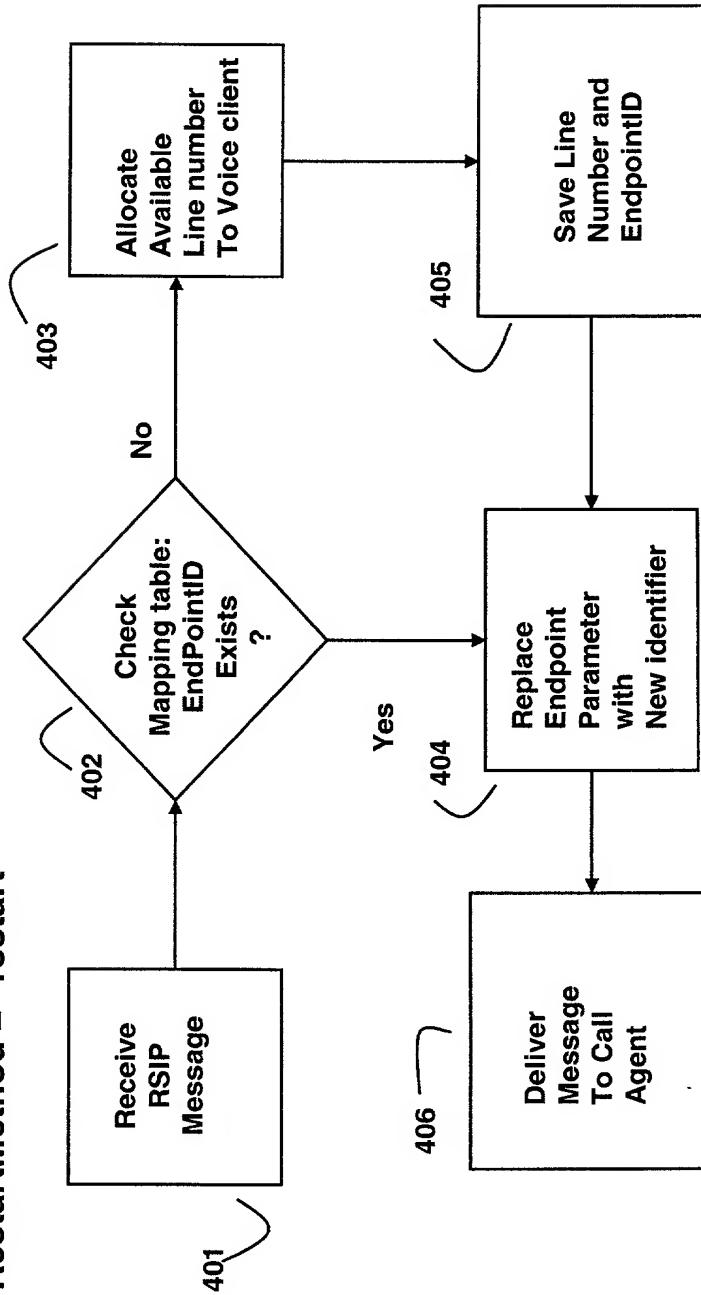
**FIG. 3**

**Client Registration ("Restart In Progress" (RSIP))  
RSIP(EndPointID, RestartMethod [, RestartDelay])**



**FIG 4a**

Restart In Progress – RSIP(EndpointID, RestartMethod [, RestartDelay])  
EndpointID = a single endpoint  
RestartMethod = “restart”



RSIP(EndpointID, RestartMethod [, RestartDelay])  
EndpointID = a single endpoint  
RestartMethod = "graceful" or "forced" or "disconnected"

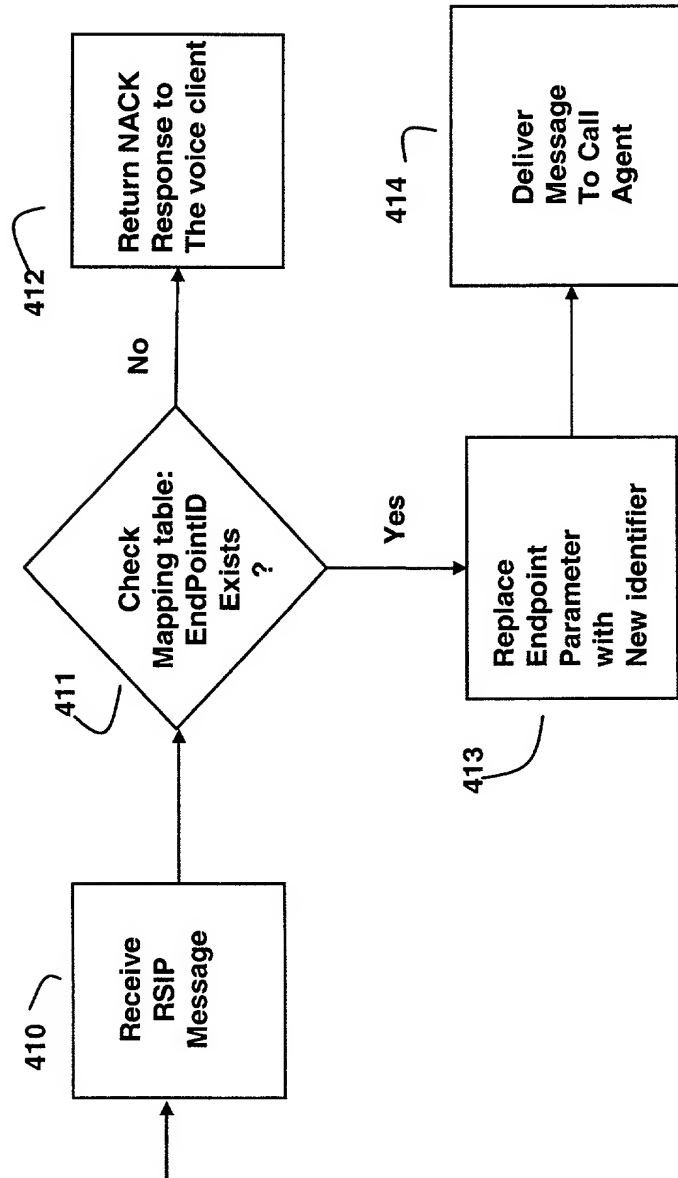
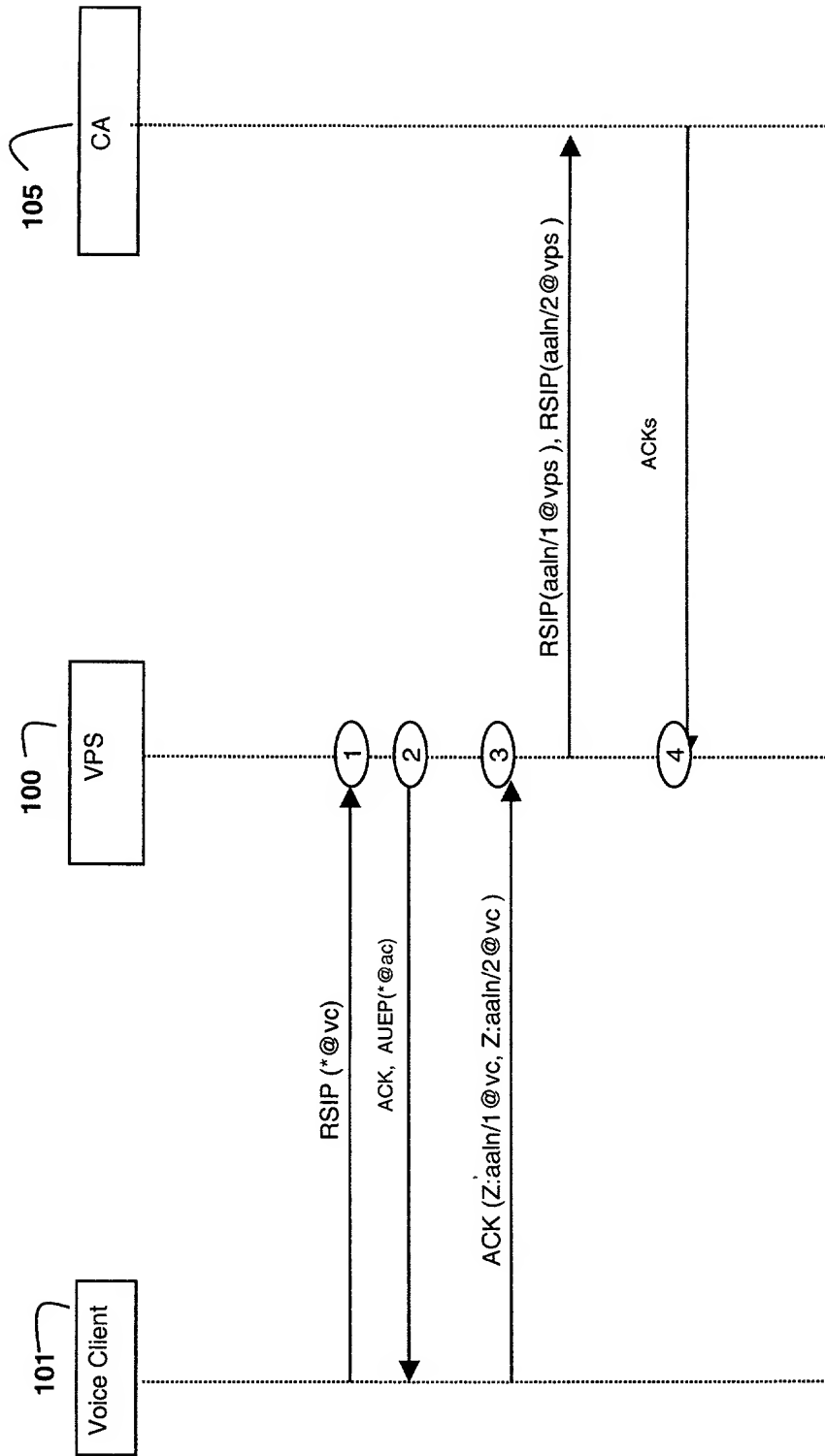


FIG. 4c

# **Client Registration ("Restart In Progress" with wild card)**



**Fig. 5a**

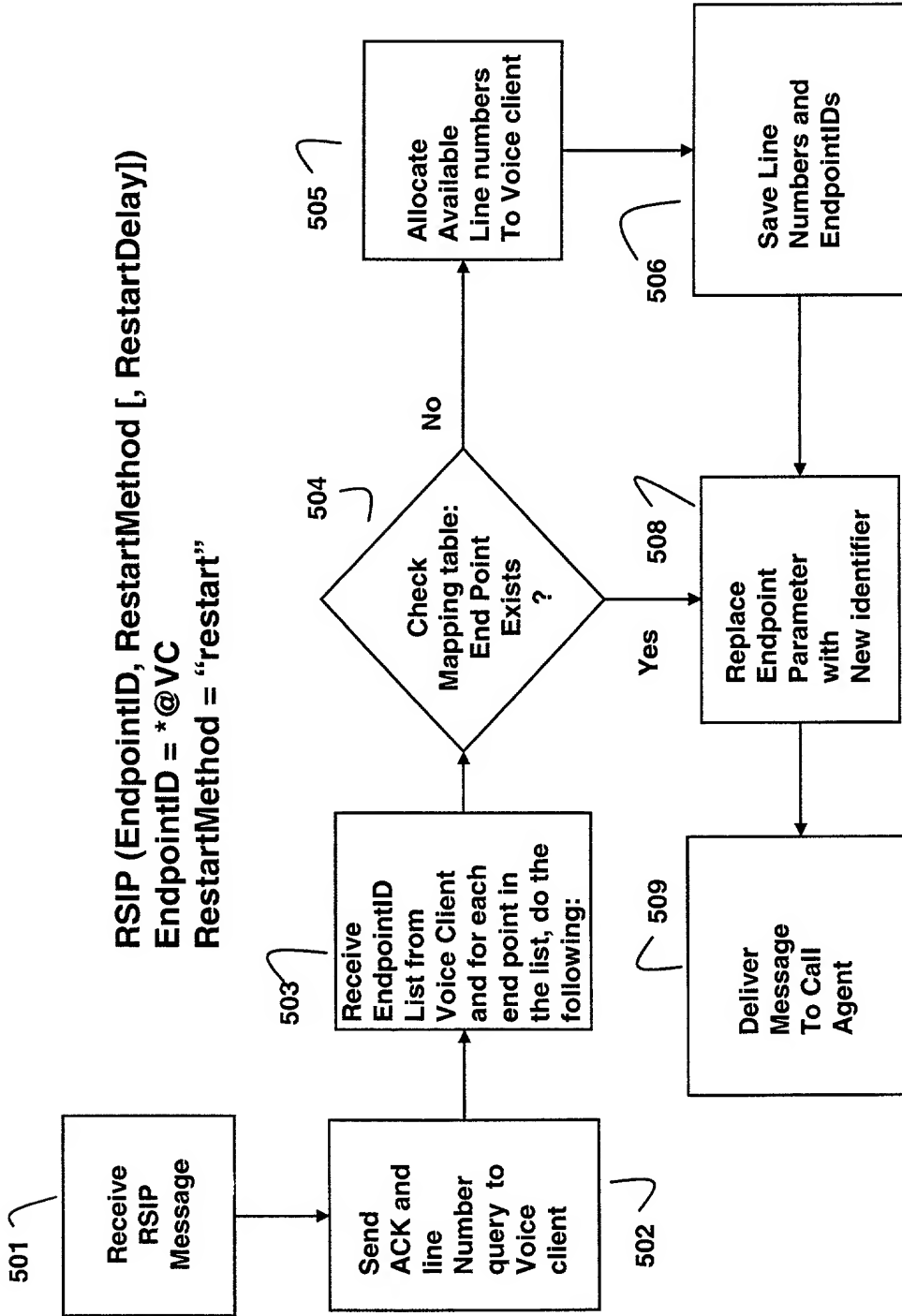


FIG. 5b



RSIP(EndpointID, RestartMethod [, RestartDelay])  
EndpointID = \*@VC  
RestartMethod = "graceful", "forced", or "disconnected"

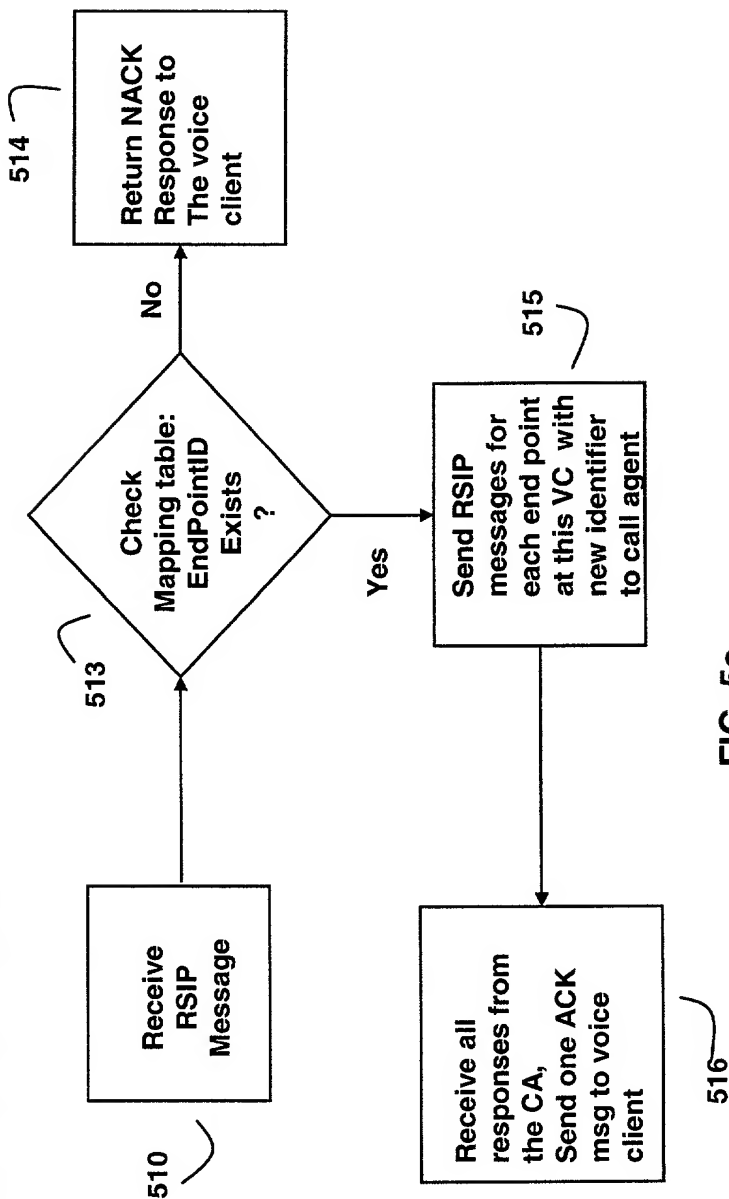
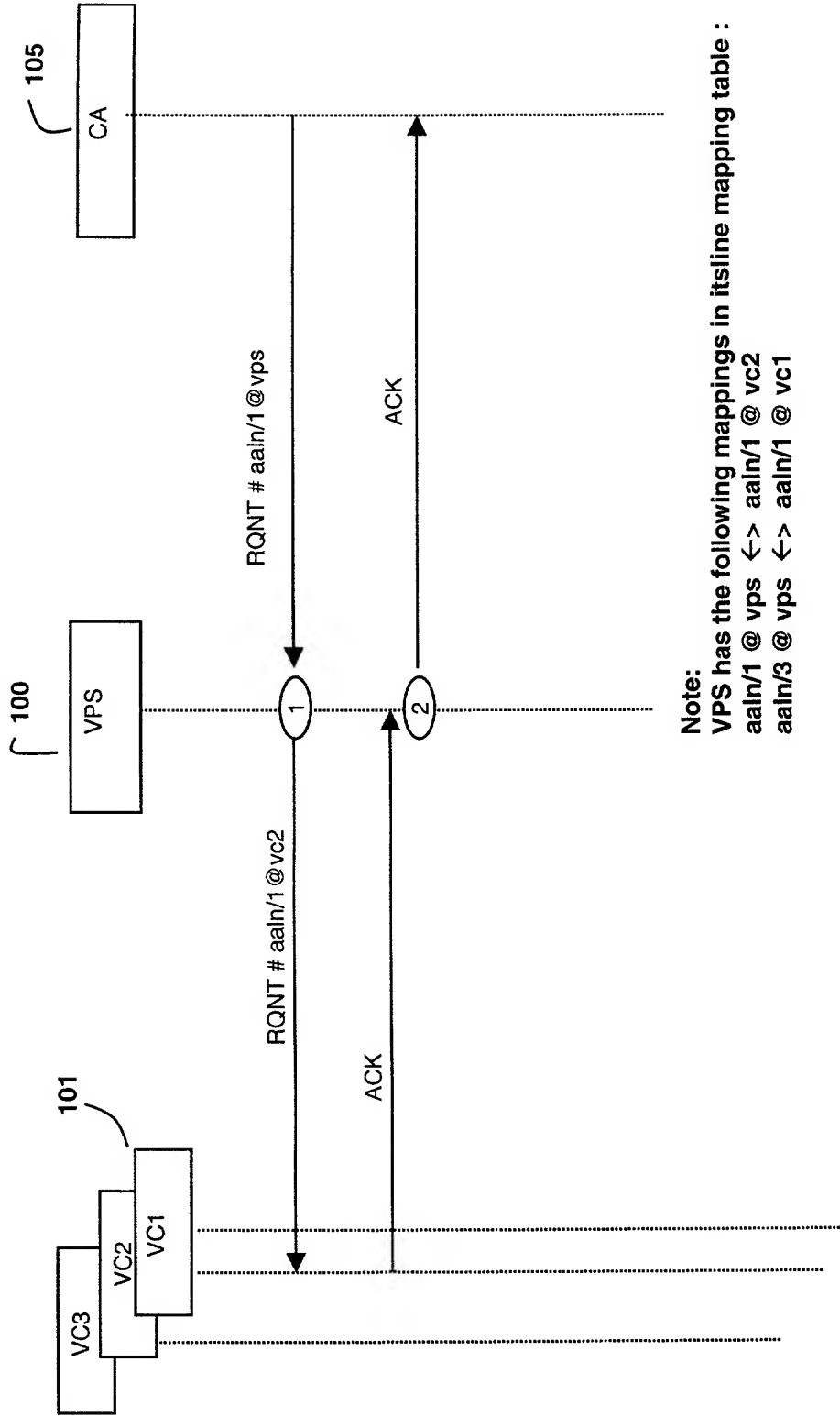


FIG. 5c

# Notification Request RQNT(EndpointID, [NotifiedEntity], ...)



Note:  
VPS has the following mappings in itsline mapping table :  
aaln/1 @ vps ↔ aaln/1 @ vc2  
aaln/3 @ vps ↔ aaln/1 @ vc1

FIG 6a

# **Notification Request (EndpointID with wild card "\*\*")**

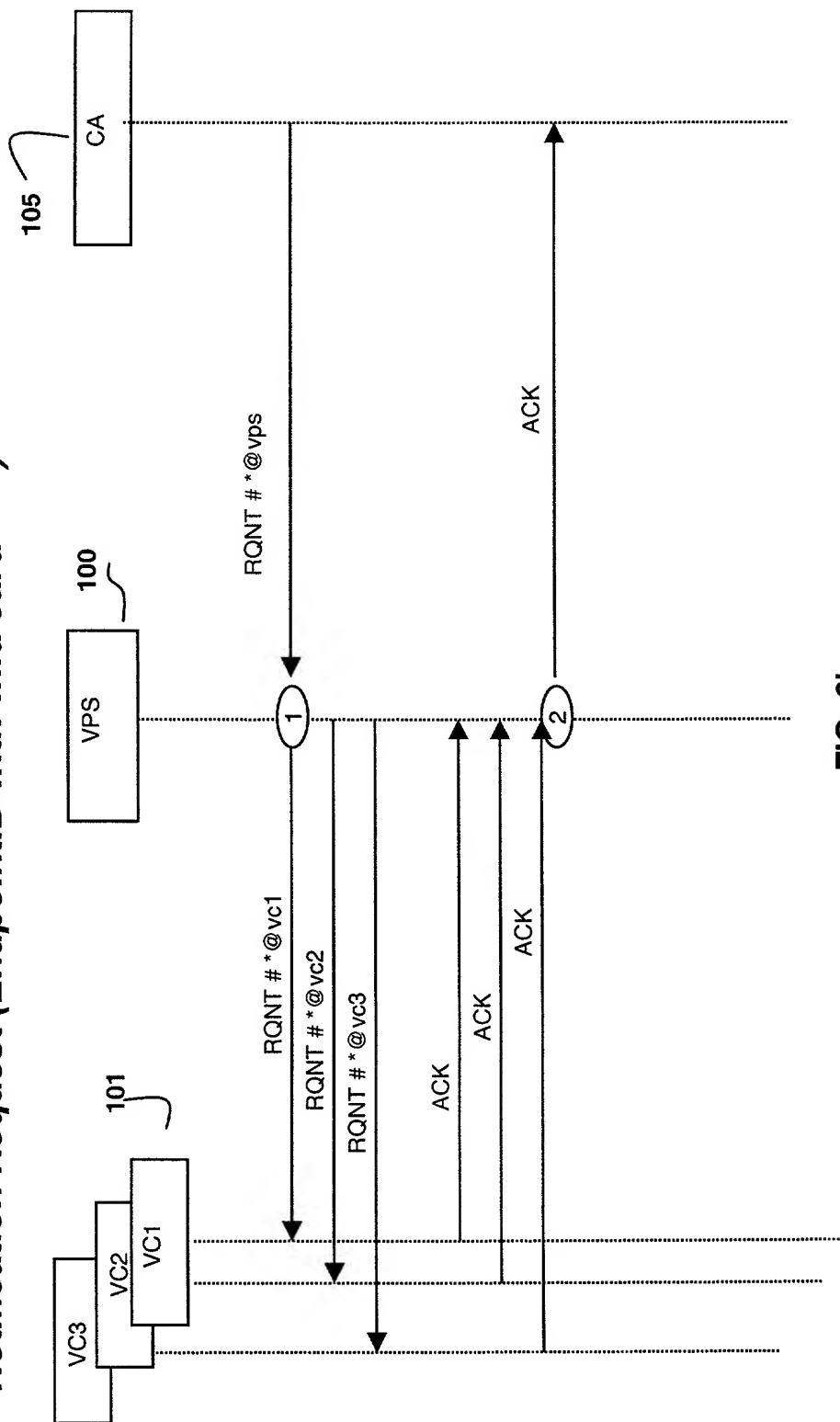
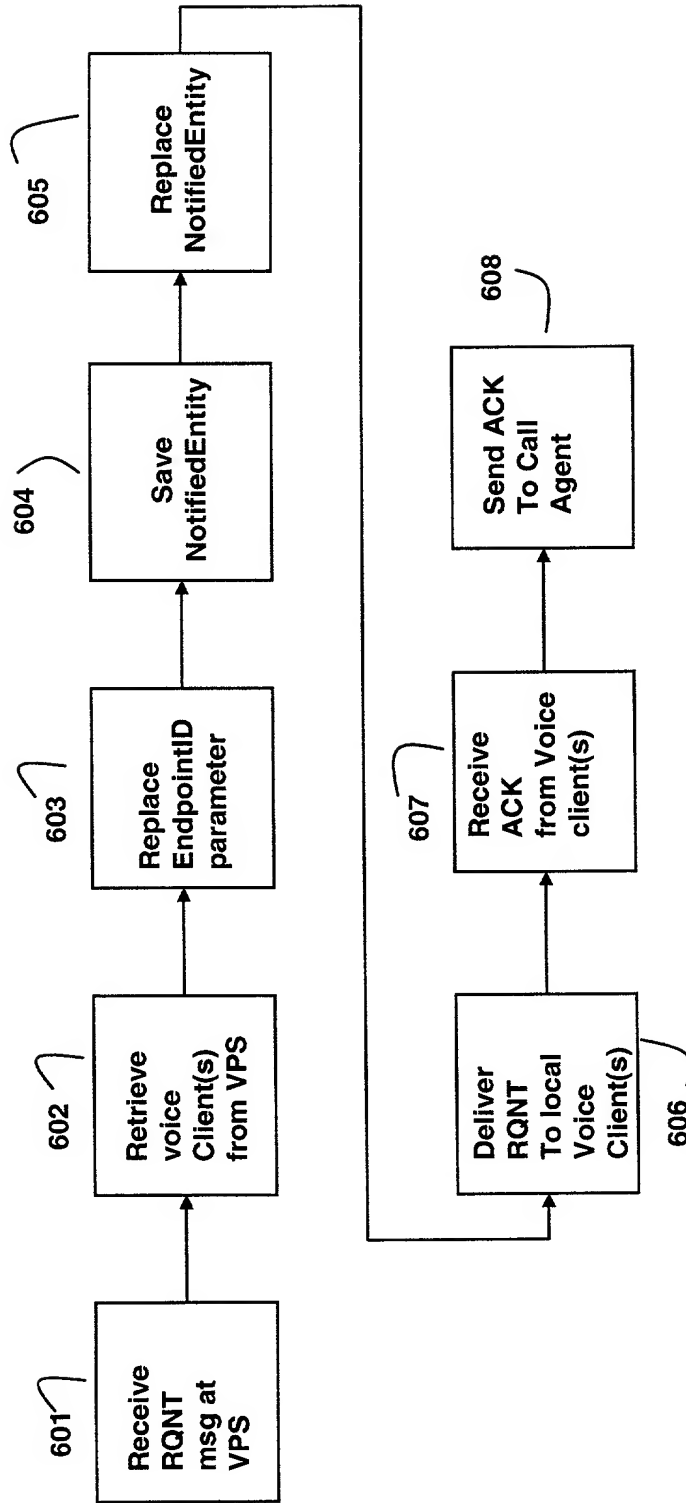


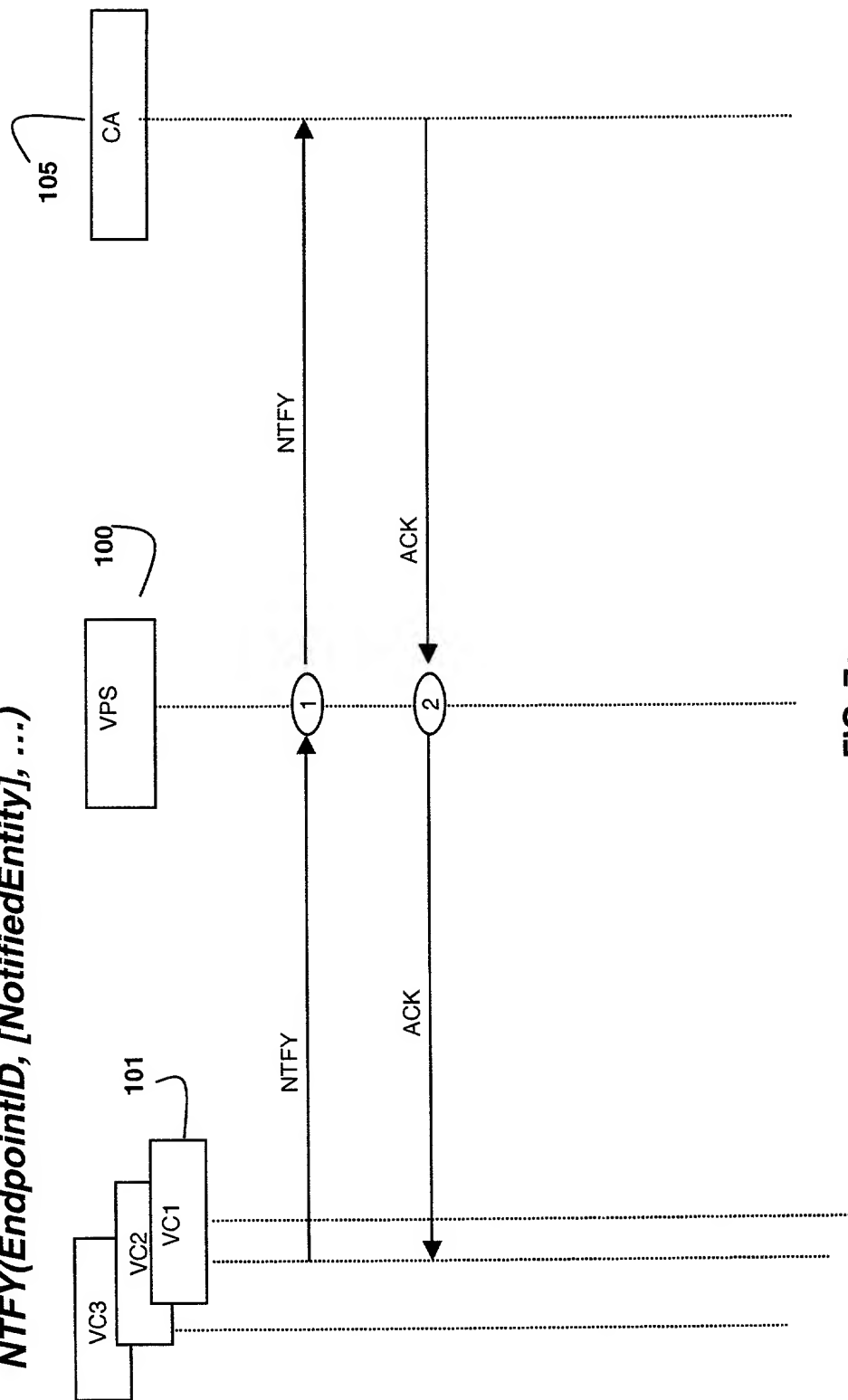
FIG. 6b

**Notification Request  
RQNT(EndpointID, [NotifiedEntity], ...)**



**FIG. 6c**

**Notifications**  
**NTFY(EndpointID, [NotifiedEntity], ...)**



**FIG. 7a**

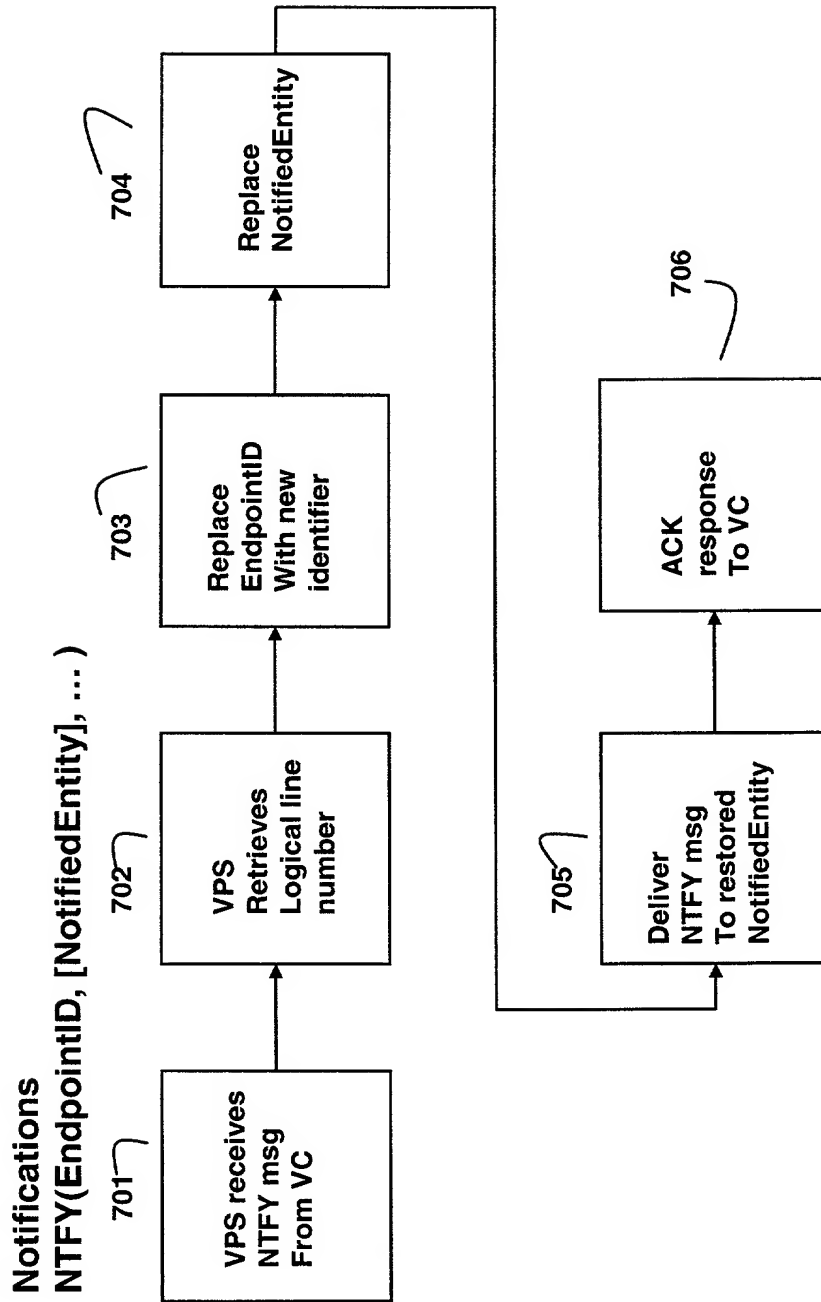


FIG. 7b

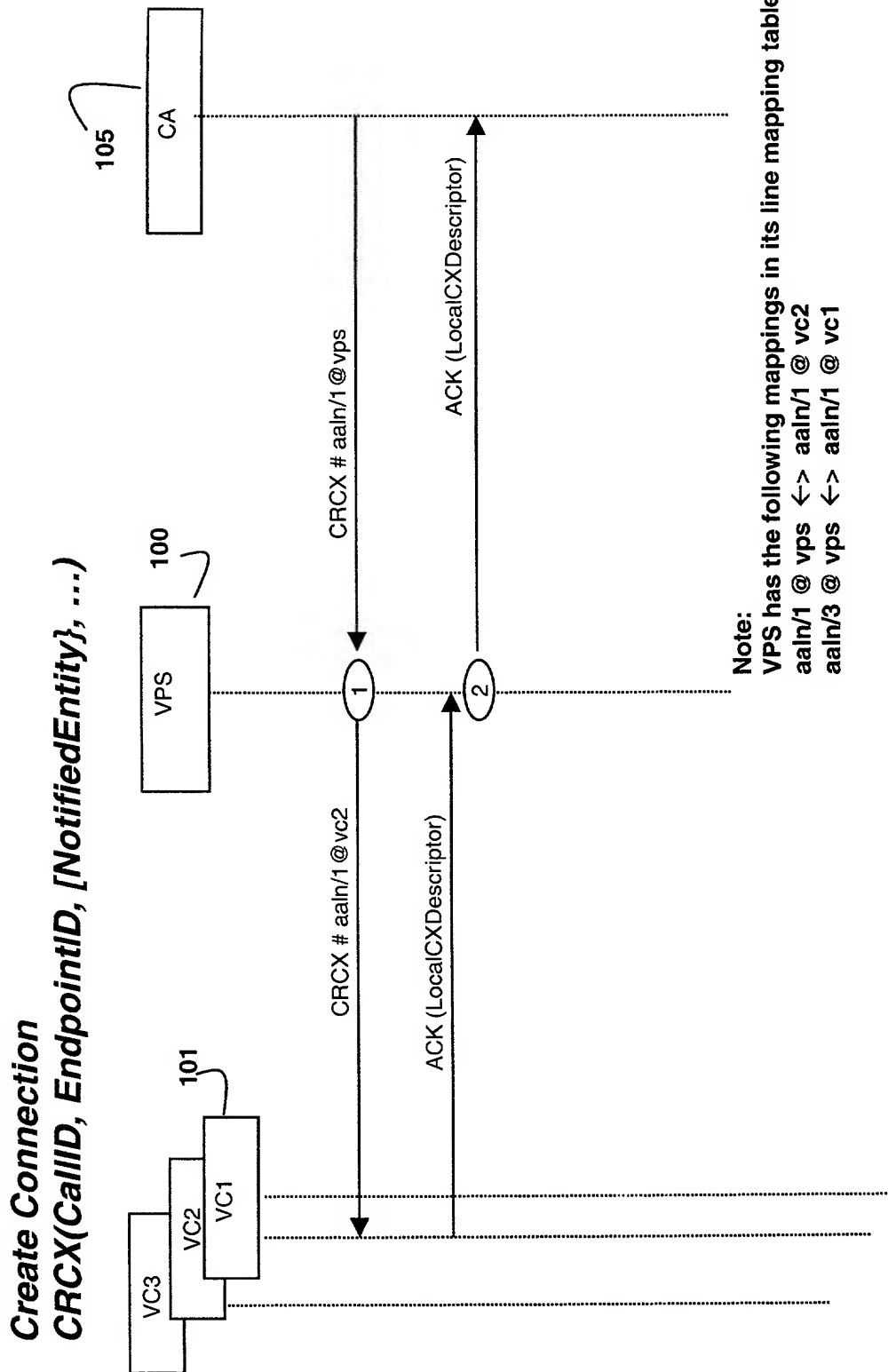


FIG. 8a

# Create Connection CRCX(CallID, EndpointID, [NotifiedEntity], ...) EndpointID = single endpoint

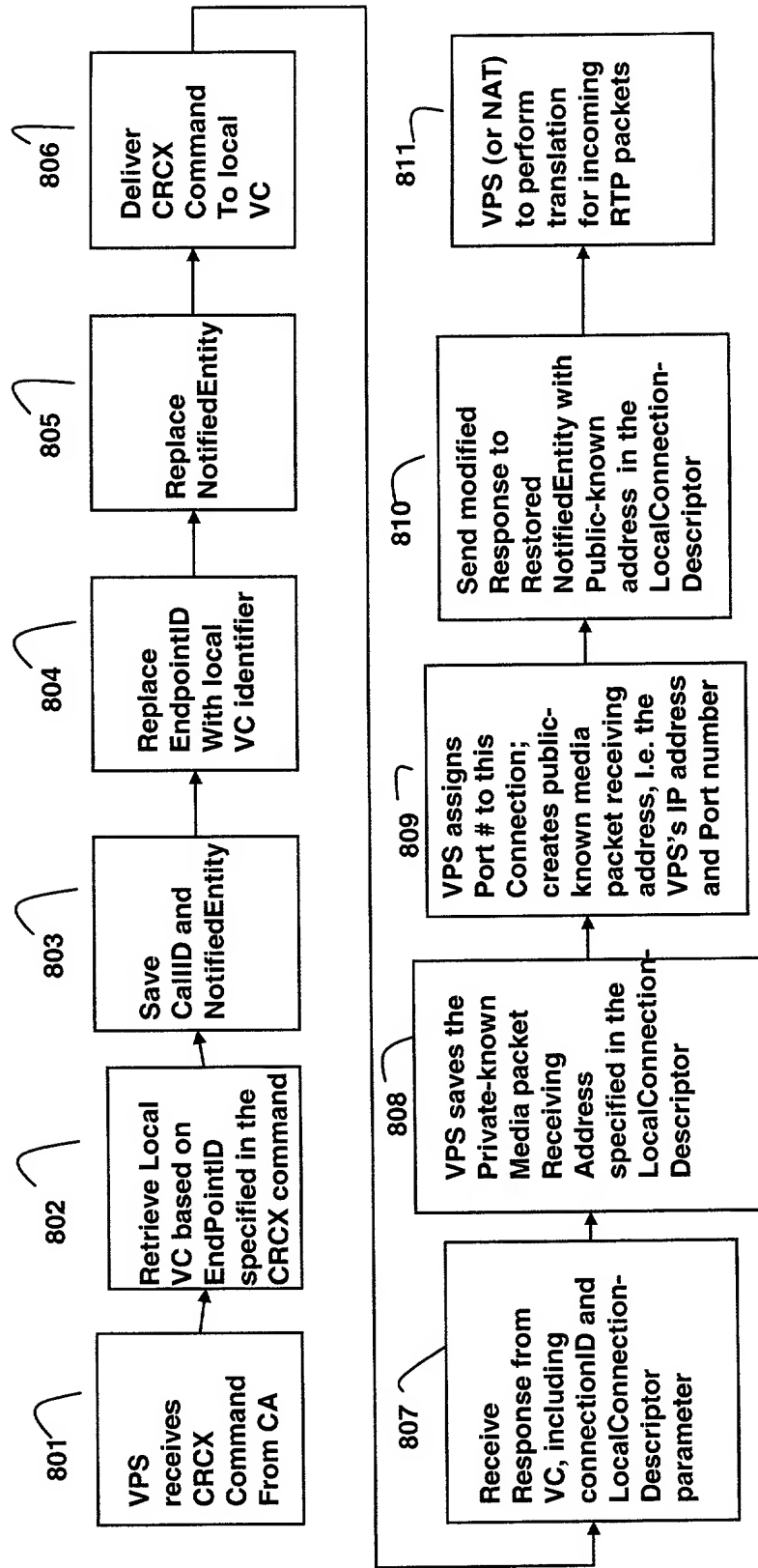
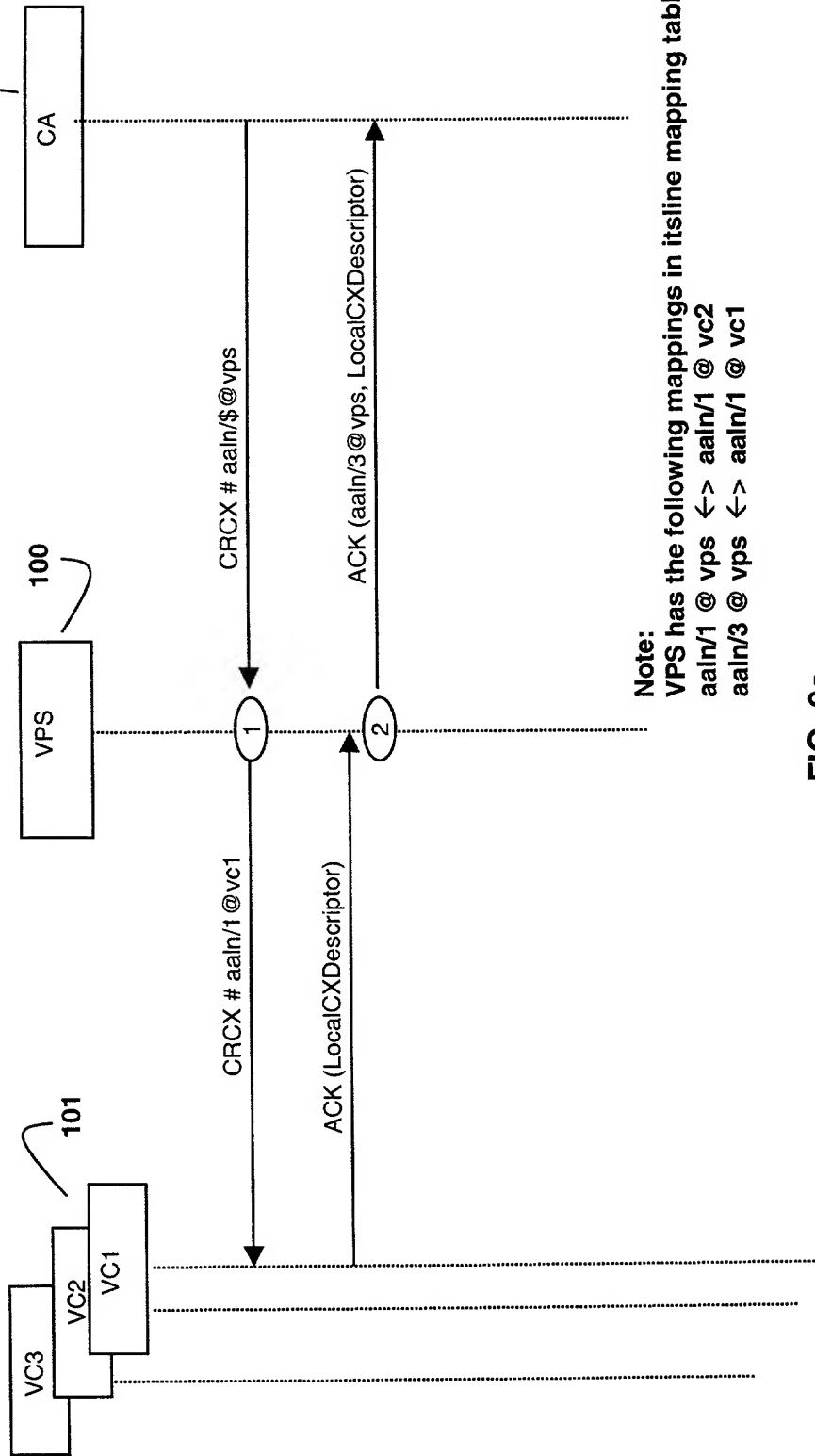


FIG. 8b



# Create Connection (EndpointId with wild card "\$" ) CRCX(CallID, EndpointID, [NotifiedEntity], ...)



Note:  
VPS has the following mappings in itsline mapping table :  
aaln/1 @ vps ↔ aaln/1 @ vc2  
aaln/3 @ vps ↔ aaln/1 @ vc1

FIG. 9a

# EndpointID with wild card (\$) ; CRCX(CallID, EndpointID, [NotifiedEntity], ...)

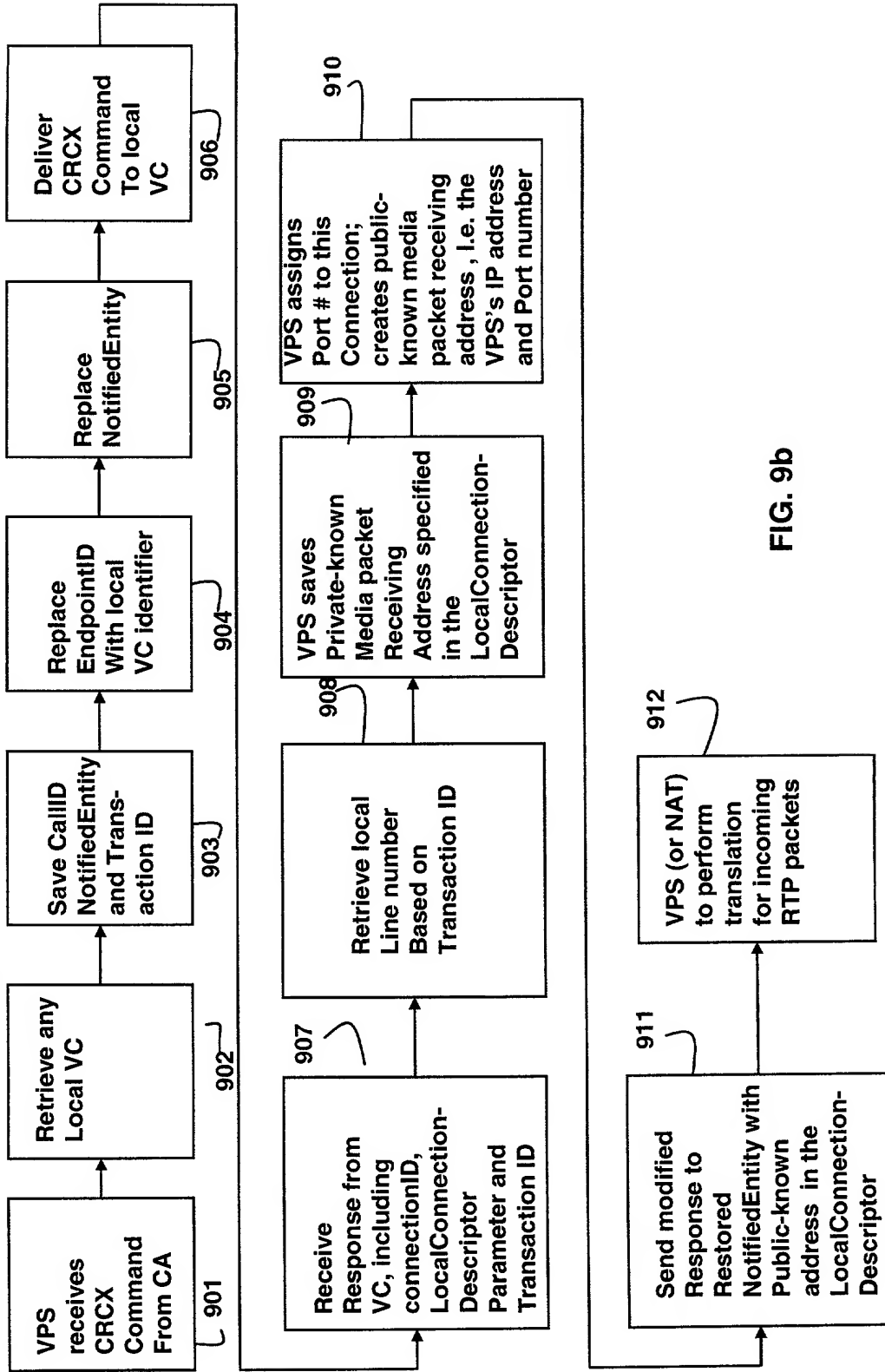
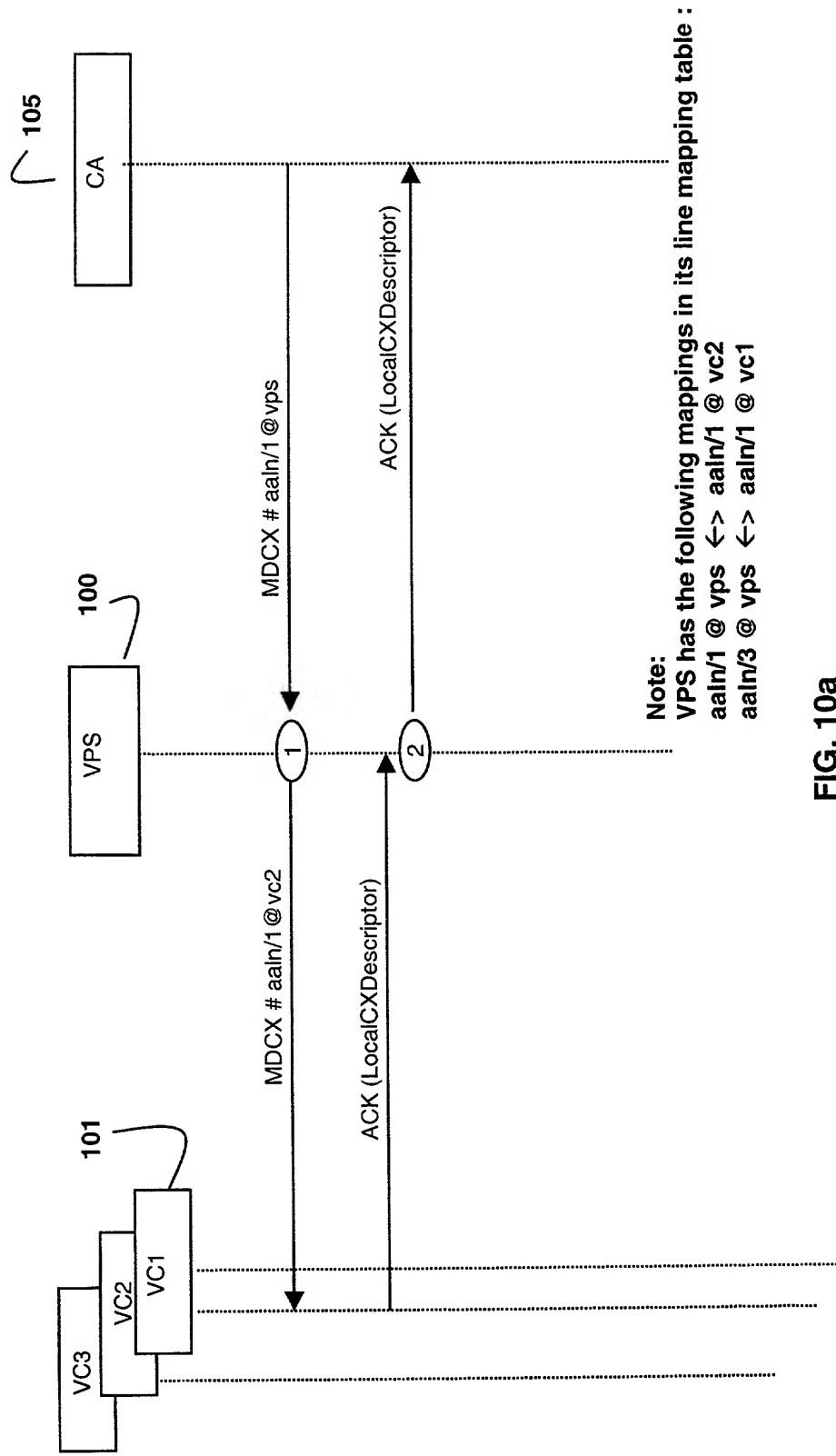


FIG. 9b

# Modify Connection

MDCX(CallID, EndpointID, ConnectionID, [NotifiedEntity], ...)



Note:  
VPS has the following mappings in its line mapping table :  
aaln/1 @ vps ↔ aaln/1 @ vc2  
aaln/3 @ vps ↔ aaln/1 @ vc1

FIG. 10a

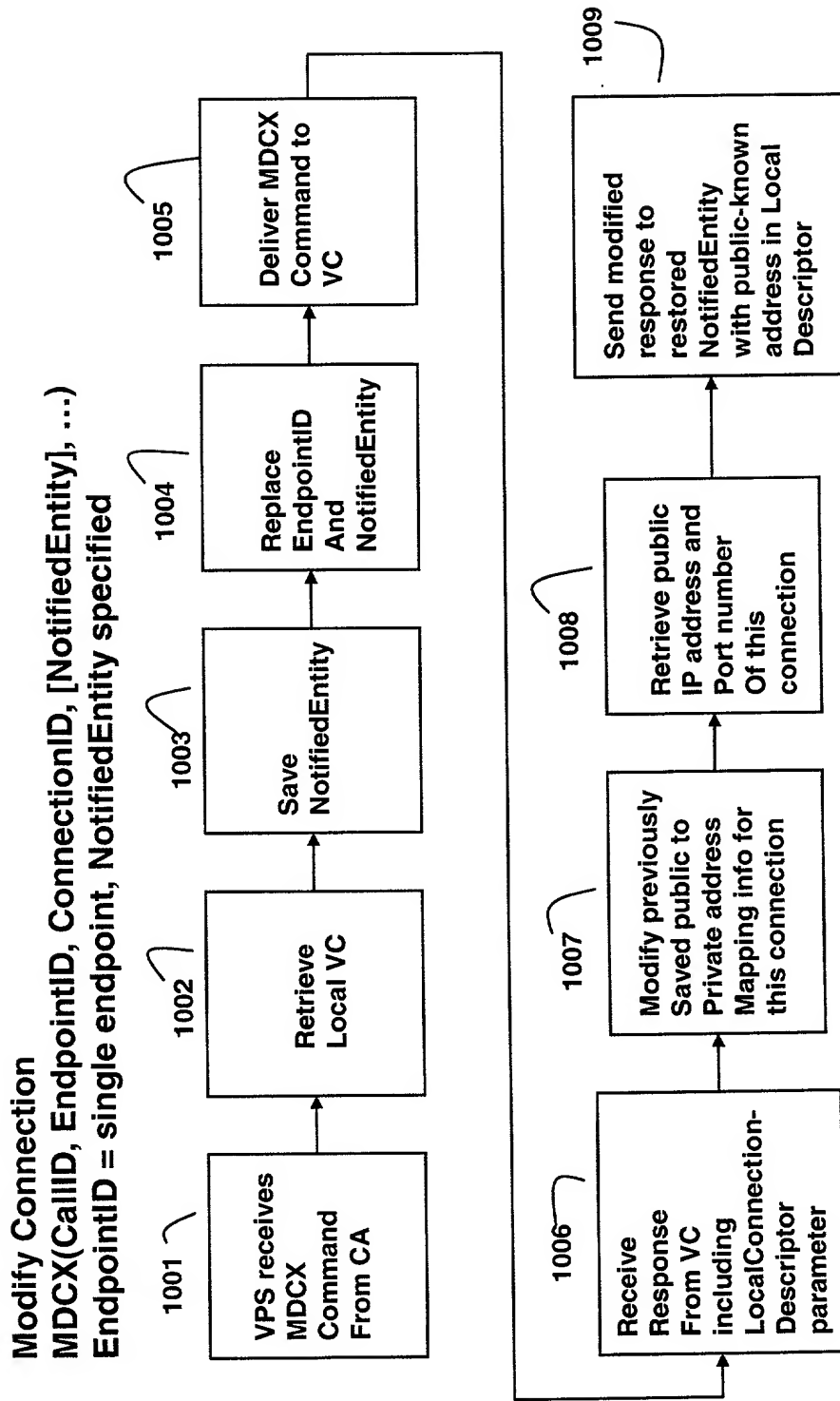
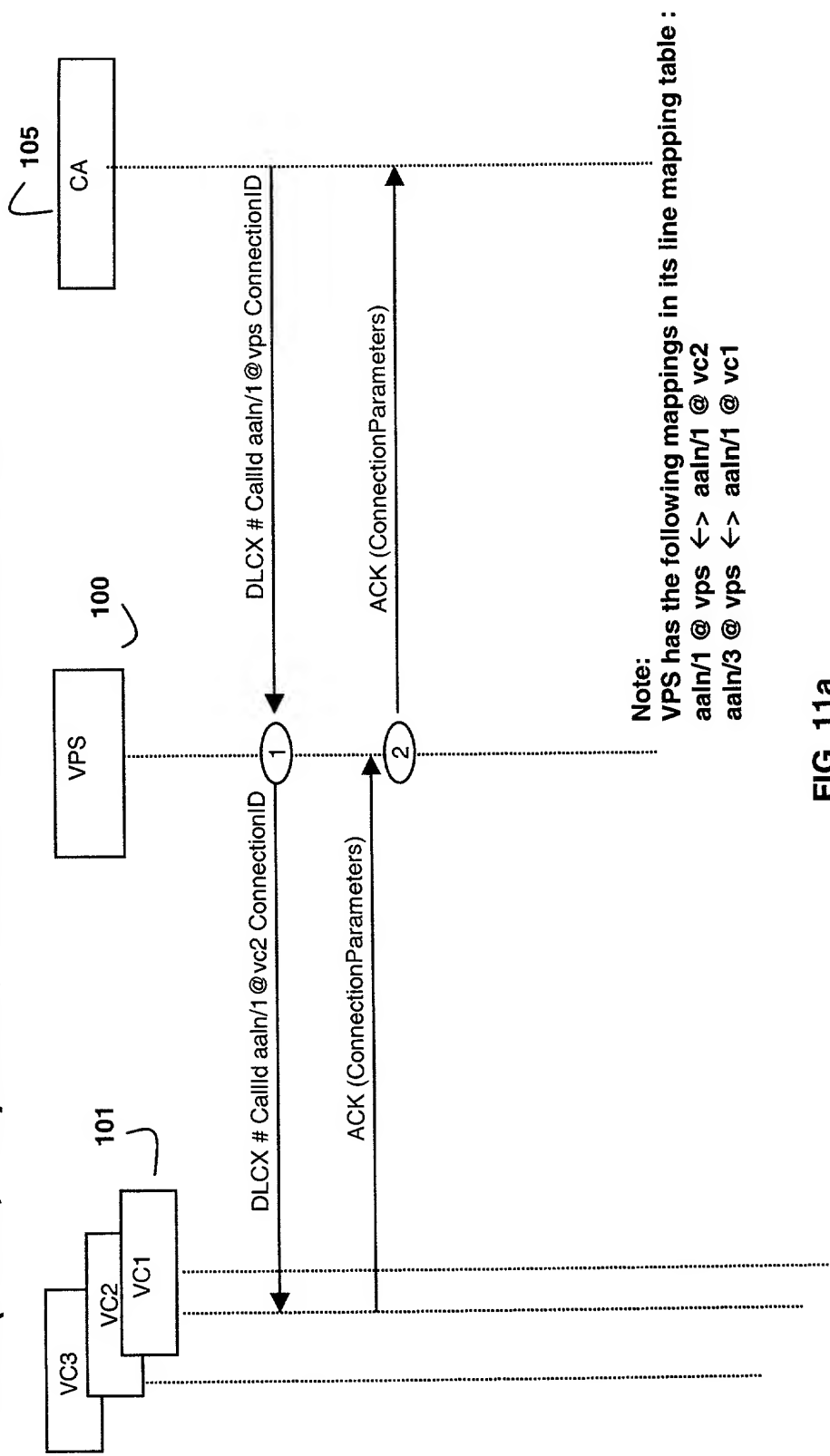


FIG. 10b

# **Delete Connection (from CA)** **DLCX(CallID, EndpointID, ConnectionID, [NotifiedEntity], ...)**



**FIG. 11a**

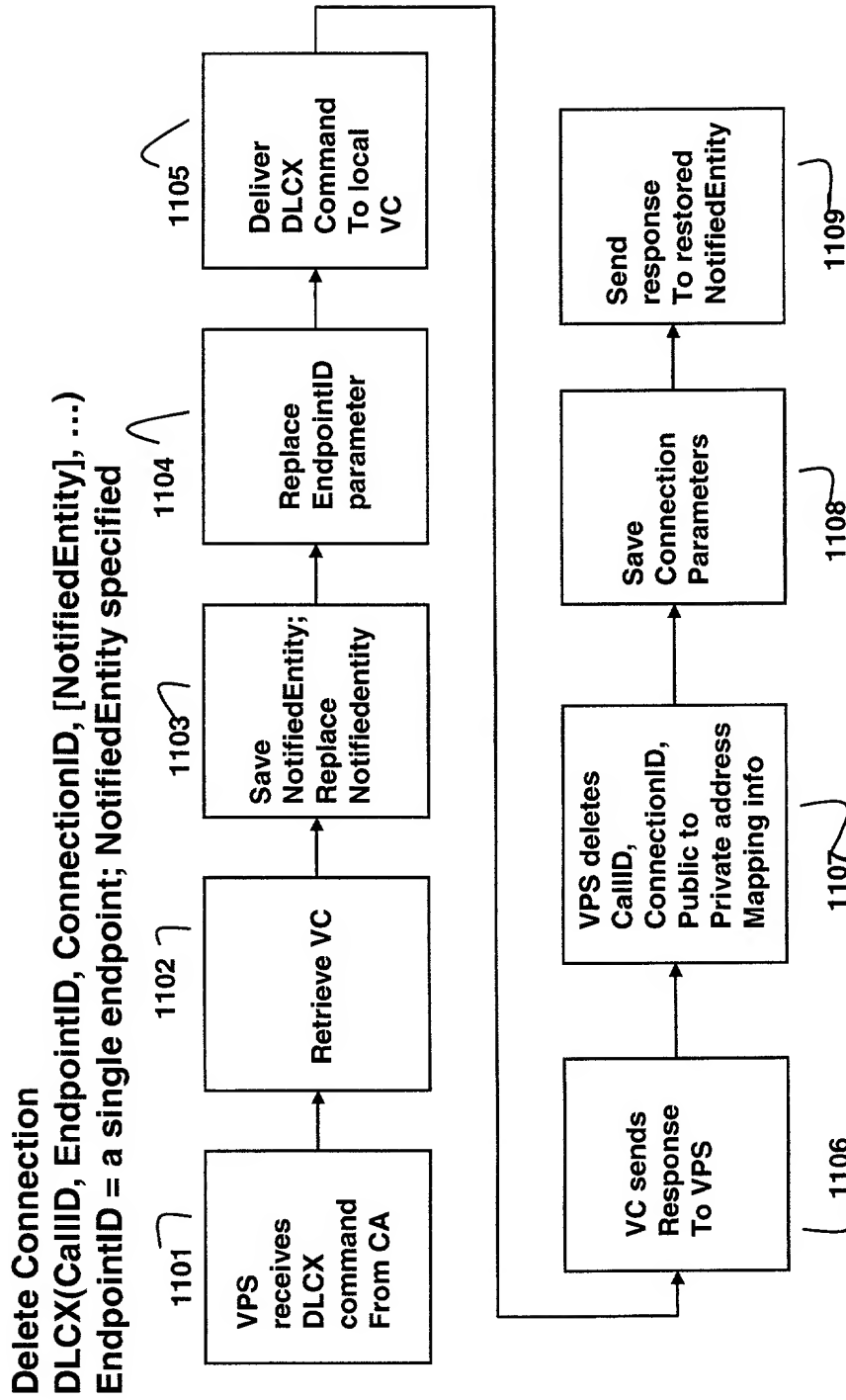


FIG. 11b

# Delete Multiple Connections (from CA: ConnectionId not Specified)

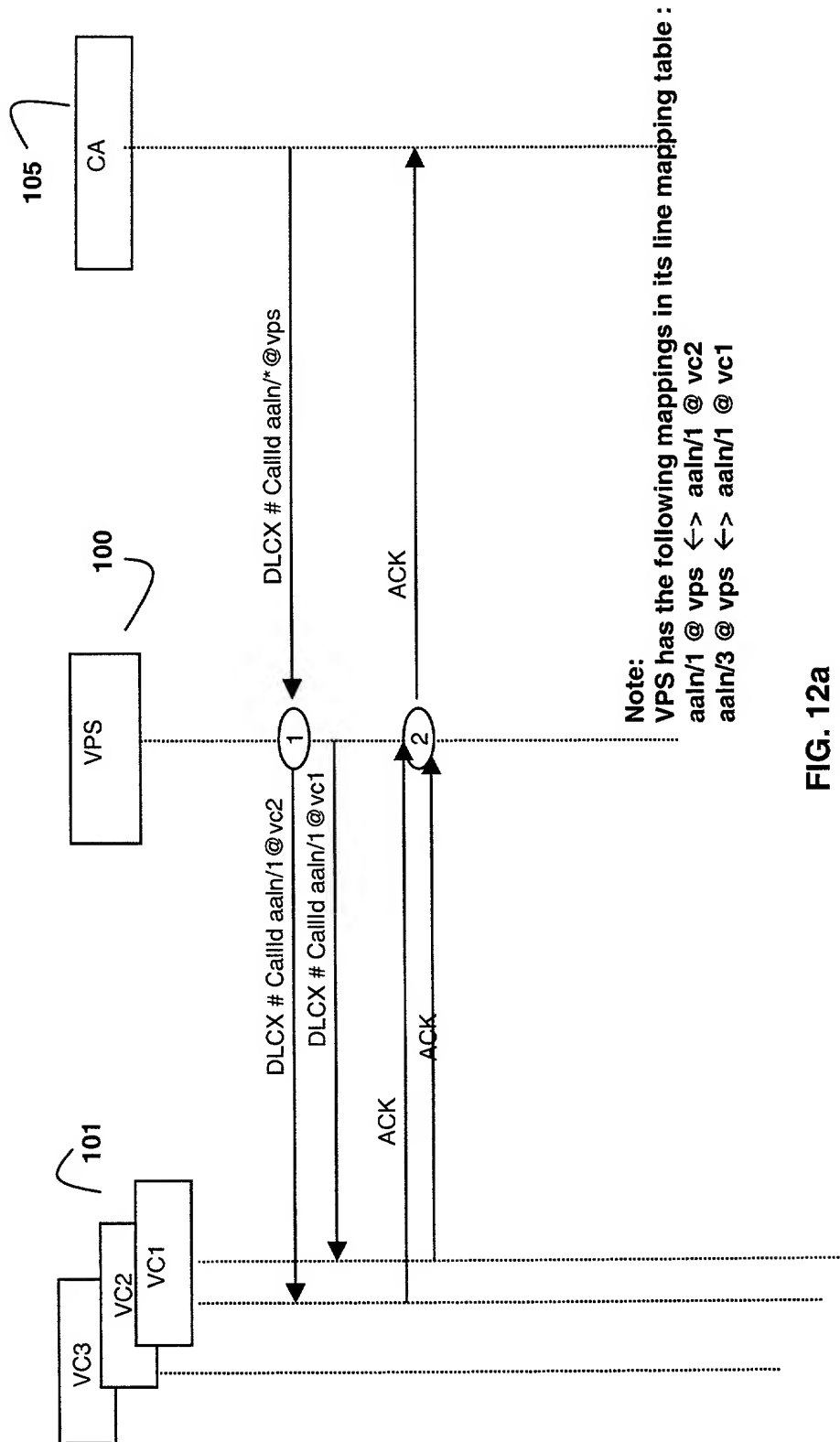


FIG. 12a

## Delete Multiple Connections DLCX(CallID, EndpointID)

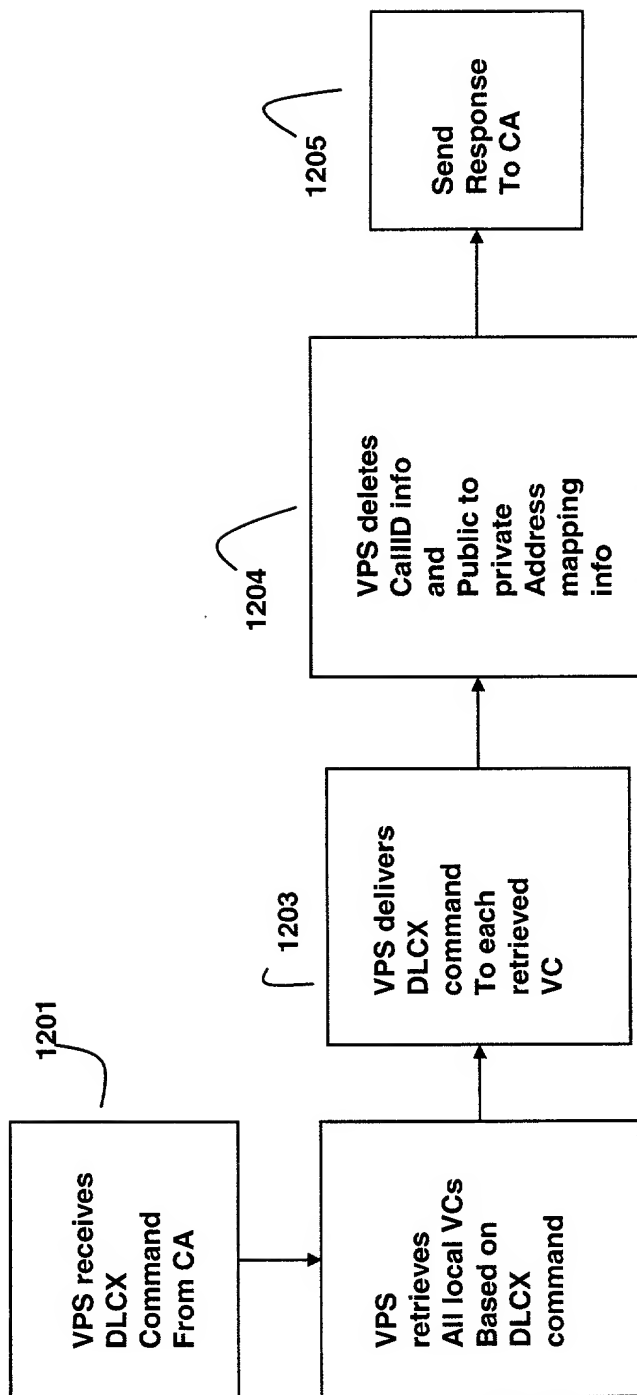


FIG. 12b



# Delete Multiple Connections (from CA: CallID & ConnectionId not Specified)

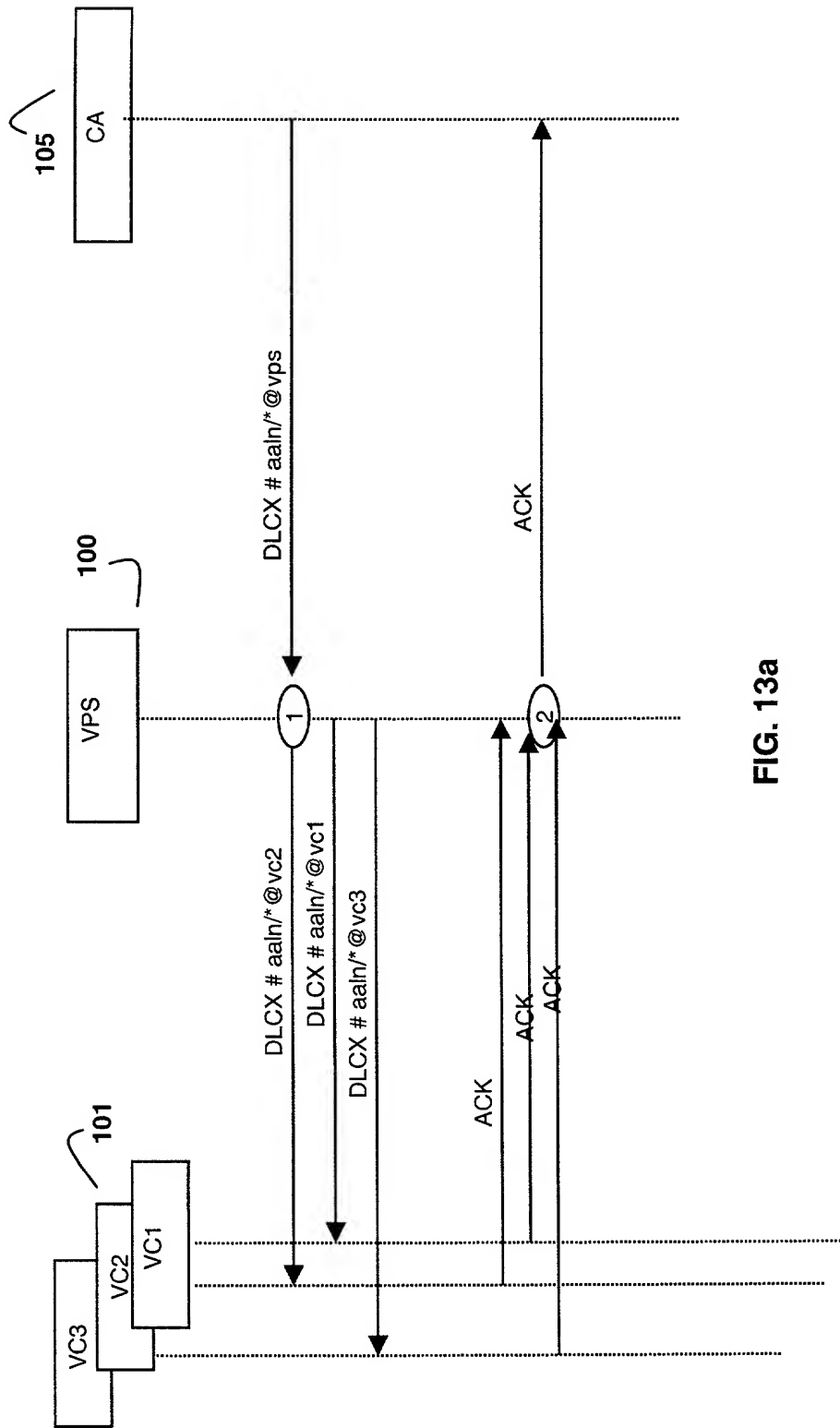
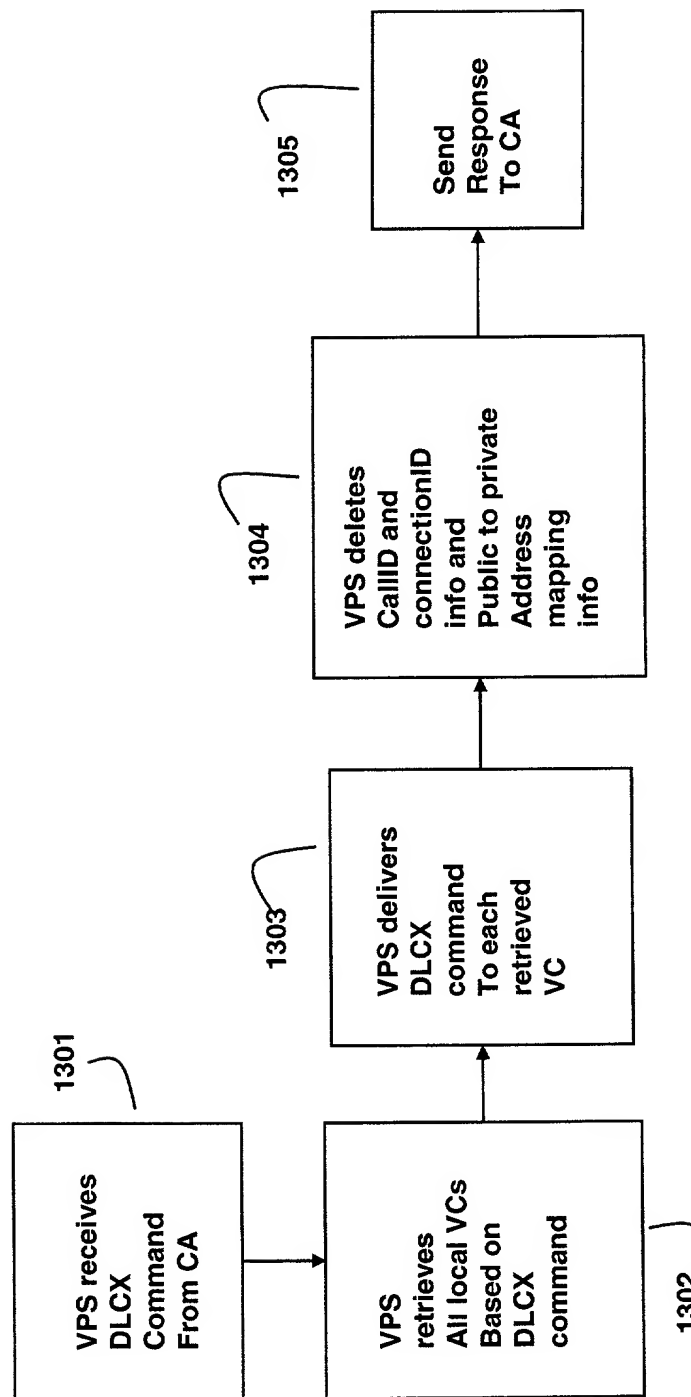


FIG. 13a

**Delete Multiple Connections  
DLCX(EndpointID)**



**FIG. 13b**

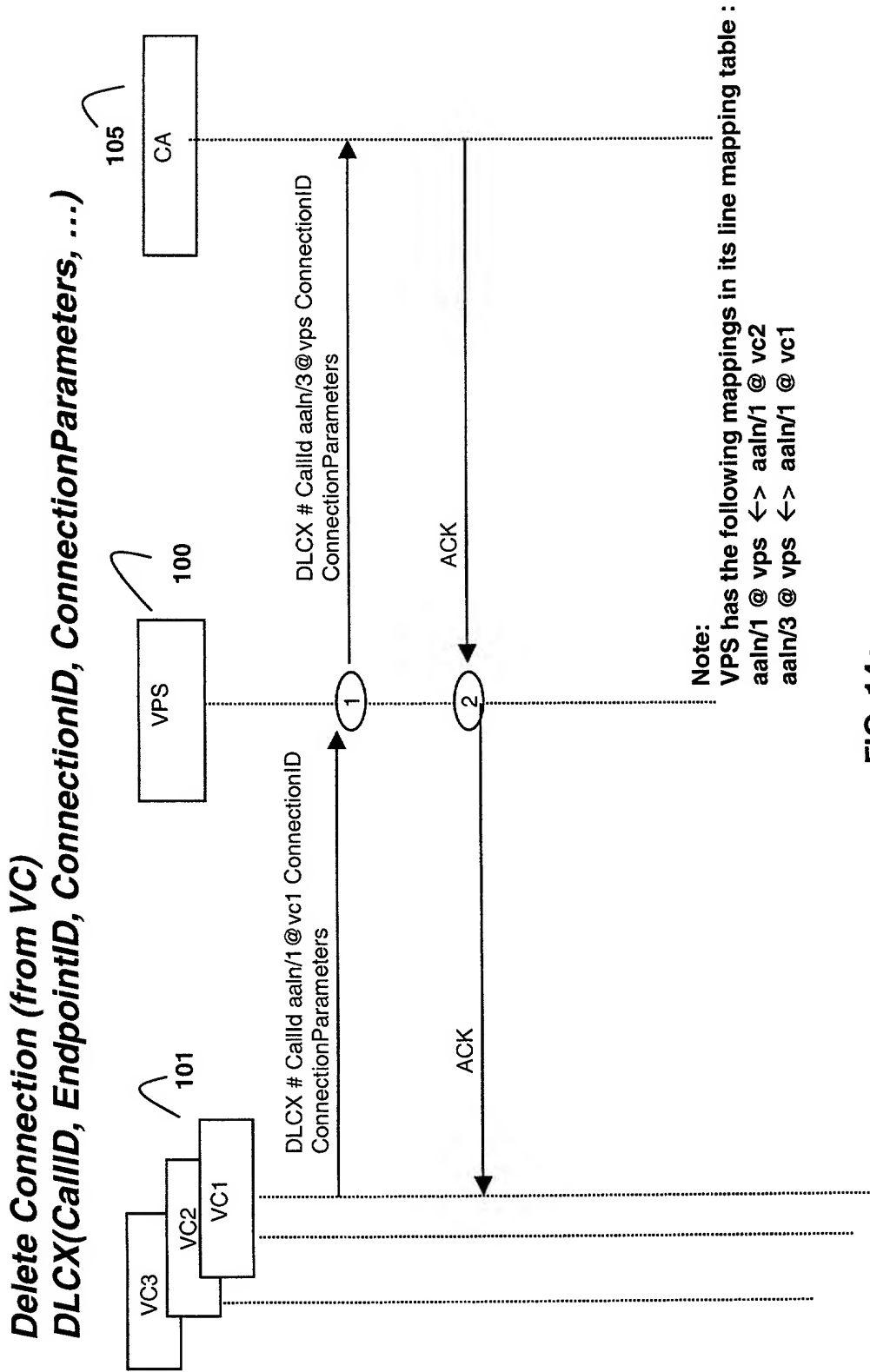


FIG. 14a

# Delete Connection DLCX(CallID, EndpointID, ConnectionID, ConnectionParameters, ...)

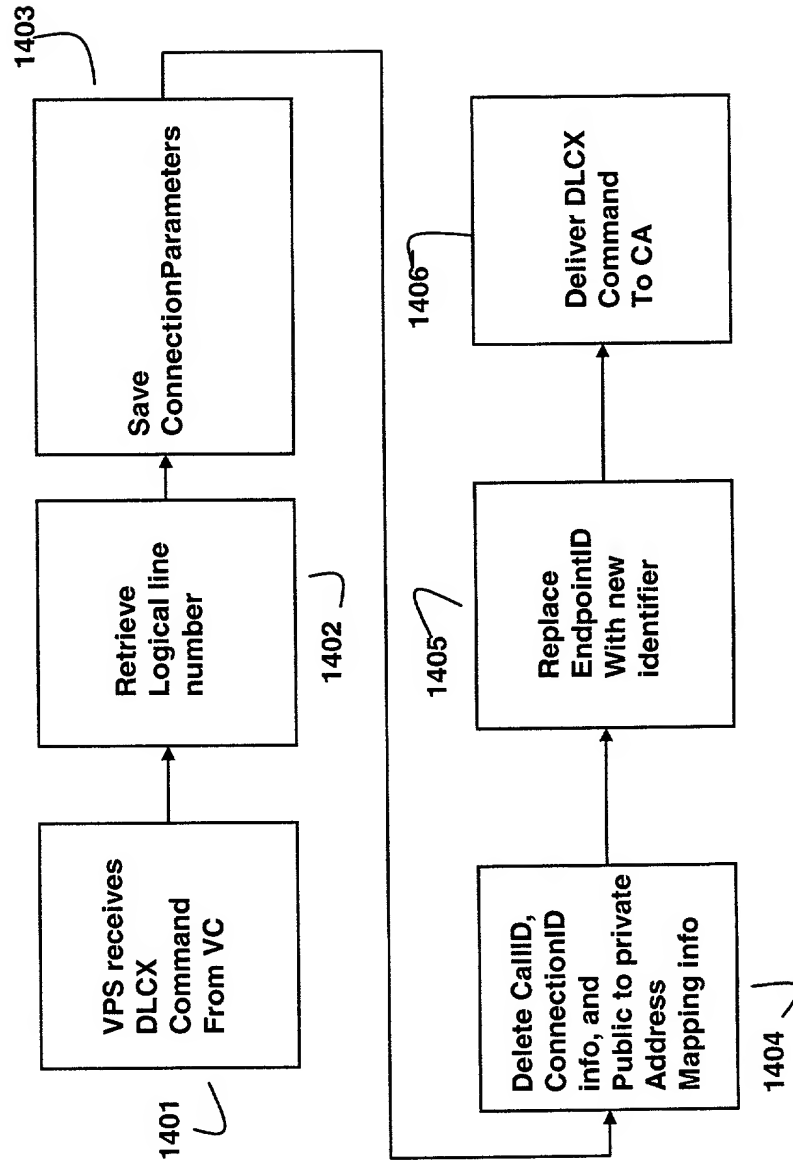


FIG. 14b

# **Audit Endpoints** **AUEP(EndpointID, [RequestedInfo])**

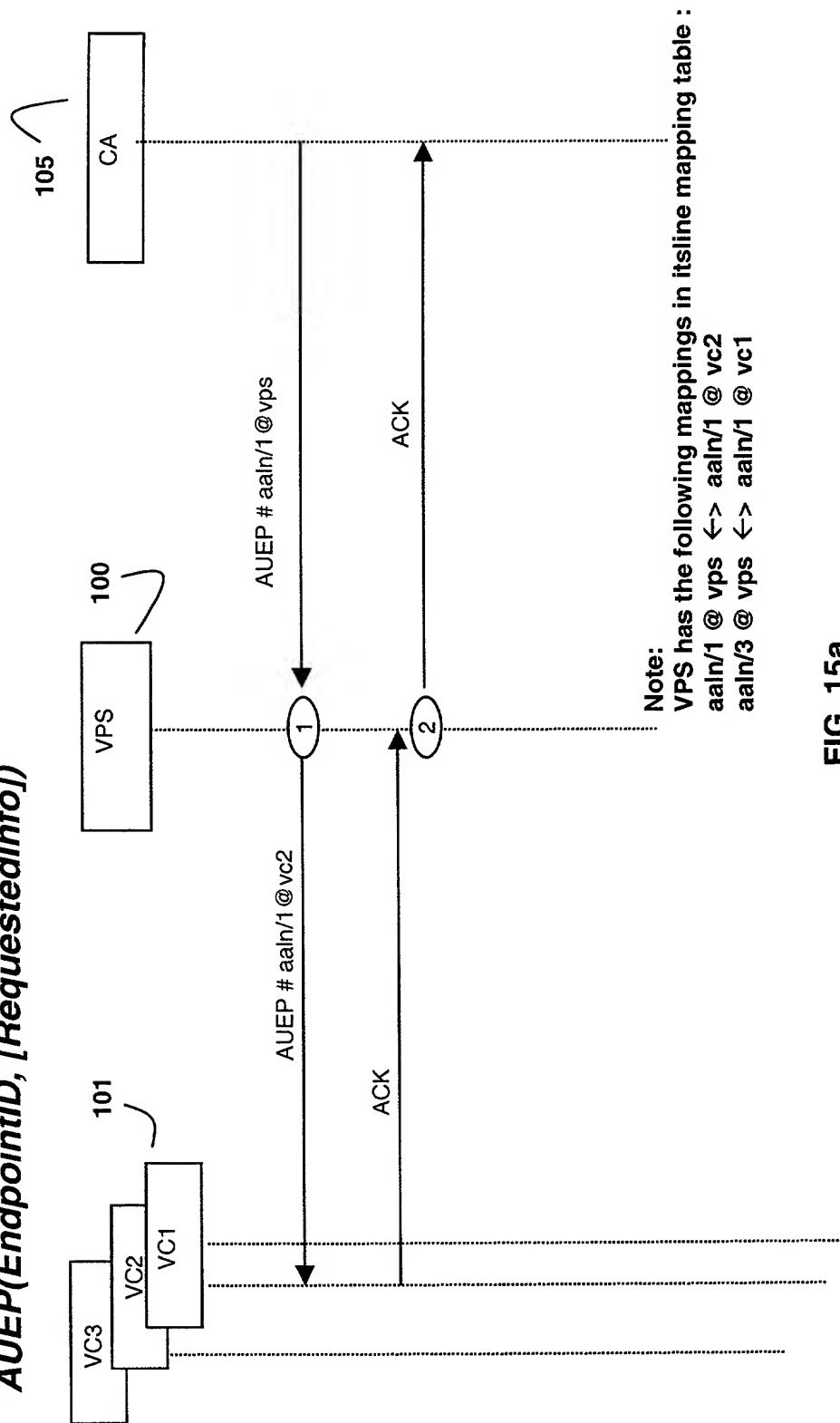
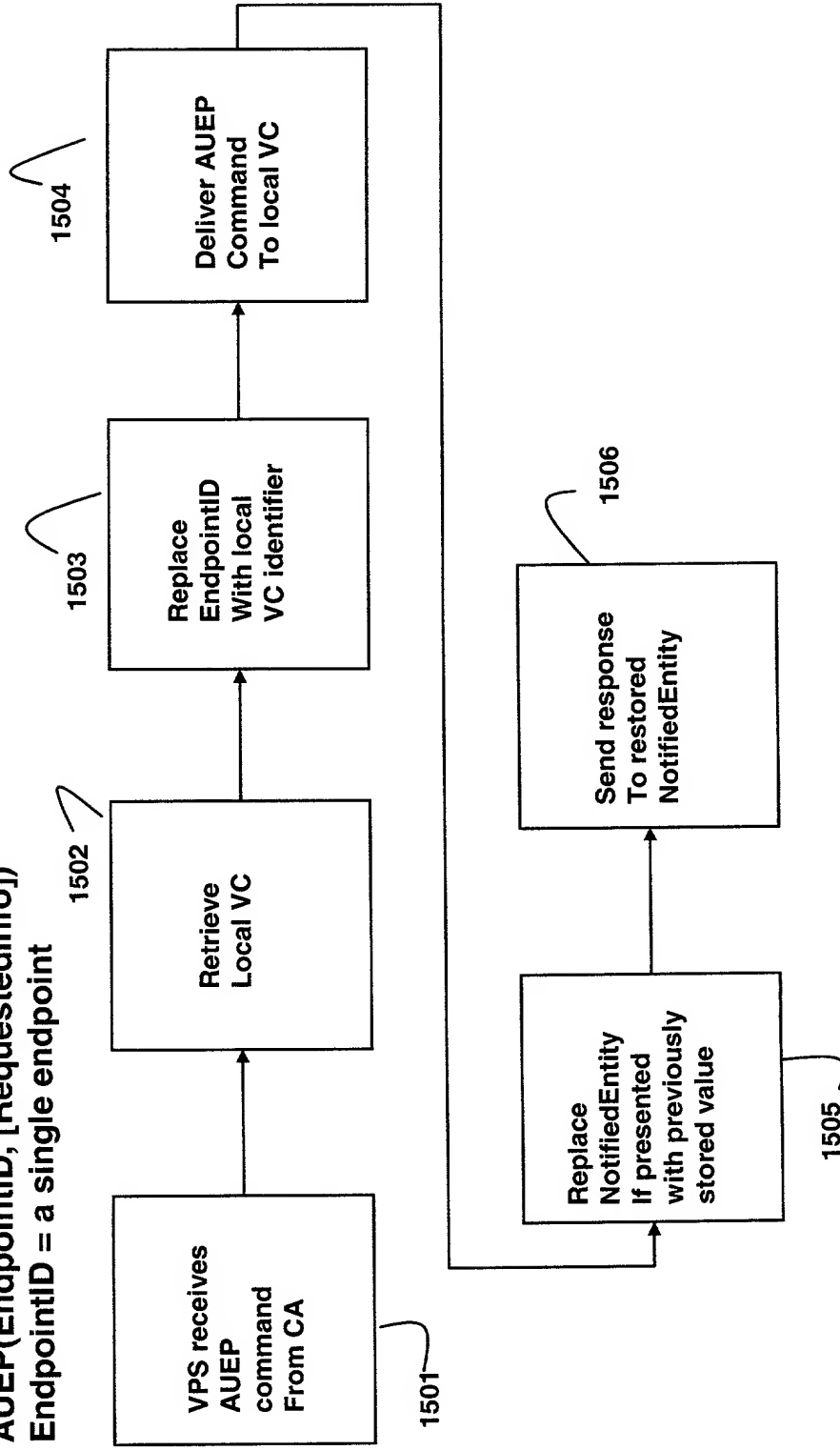


FIG. 15a

**Audit Endpoints**  
**AUEP(EndpointID, [RequestedInfo])**  
**EndpointID = a single endpoint**



**FIG. 15b**

# **Audit Multiple Endpoints (EndpointId with wild card “\*” )**

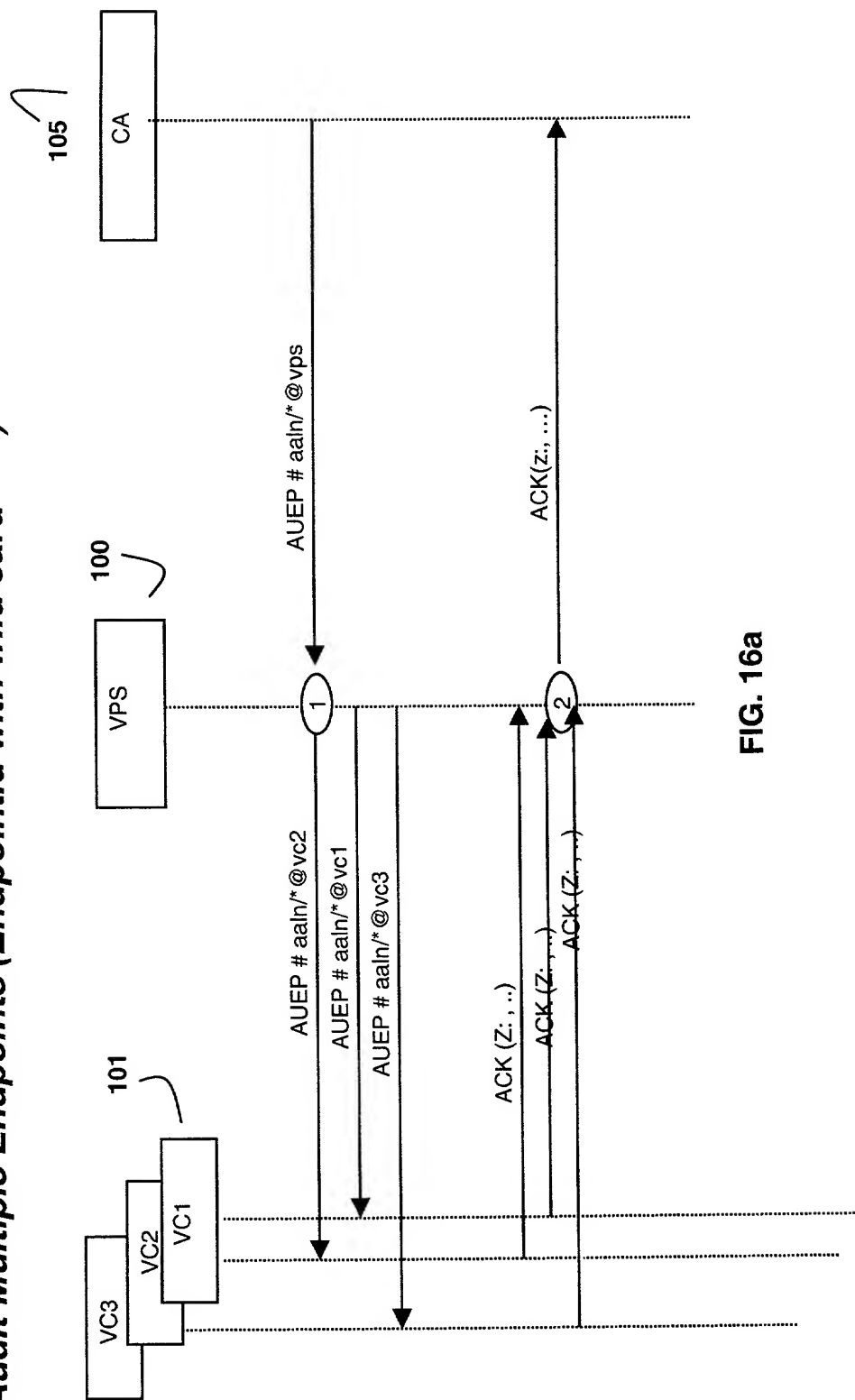
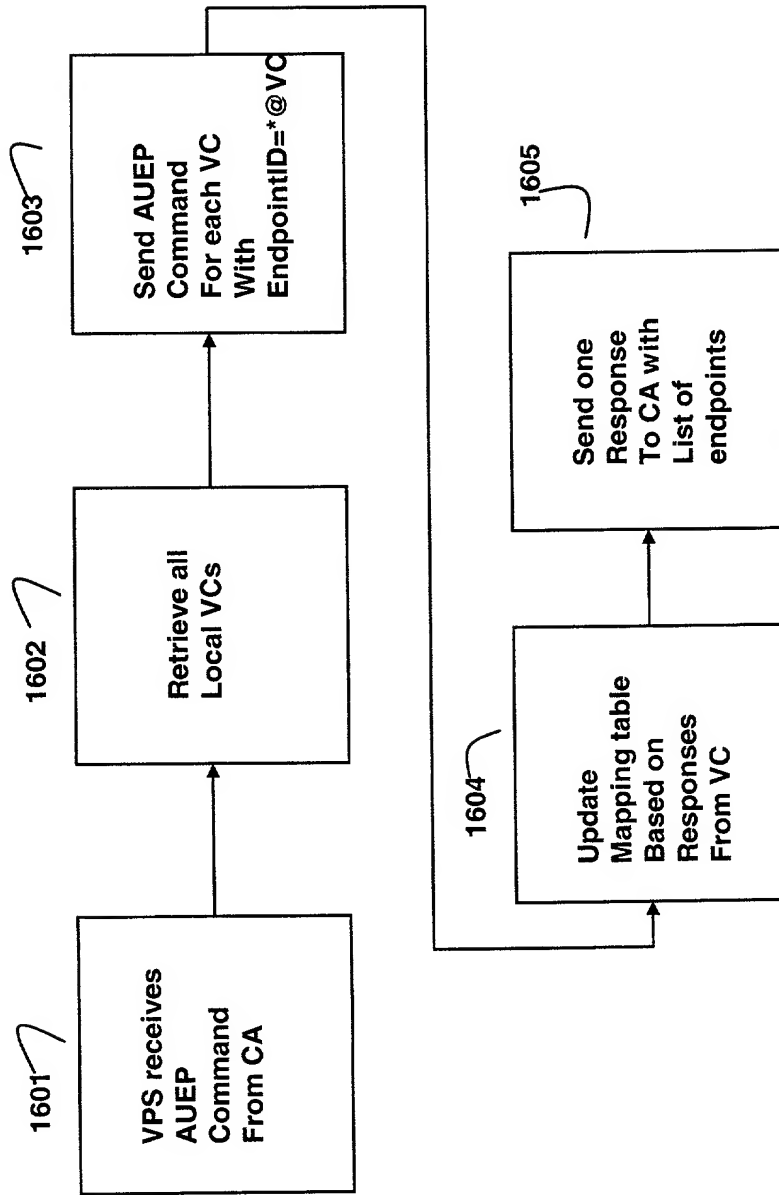


FIG. 16a

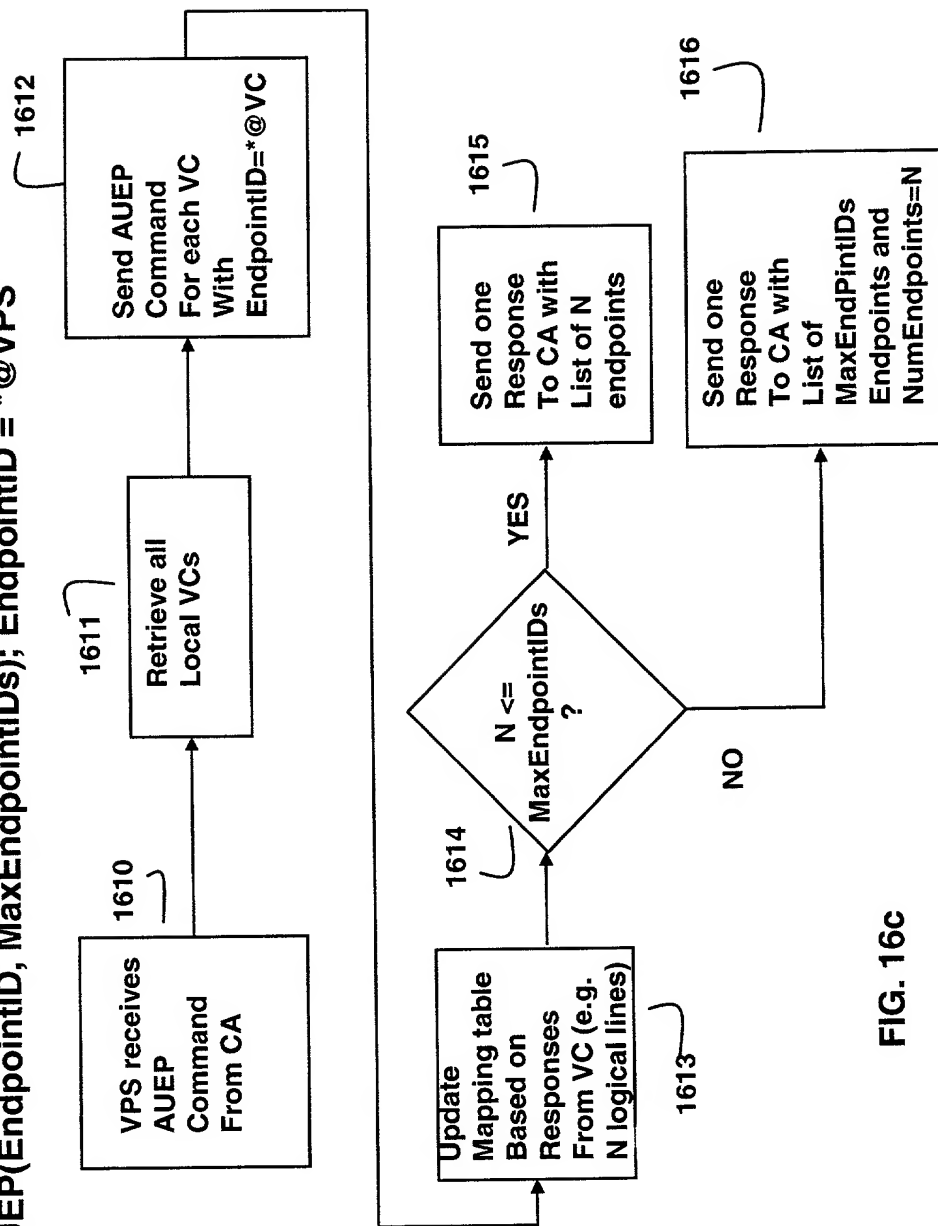
**Audit Multiple Endpoints; EndpointID with wild card  
AUEP(EndpointID); EndpointID = \*@VPS**



**FIG. 16b**

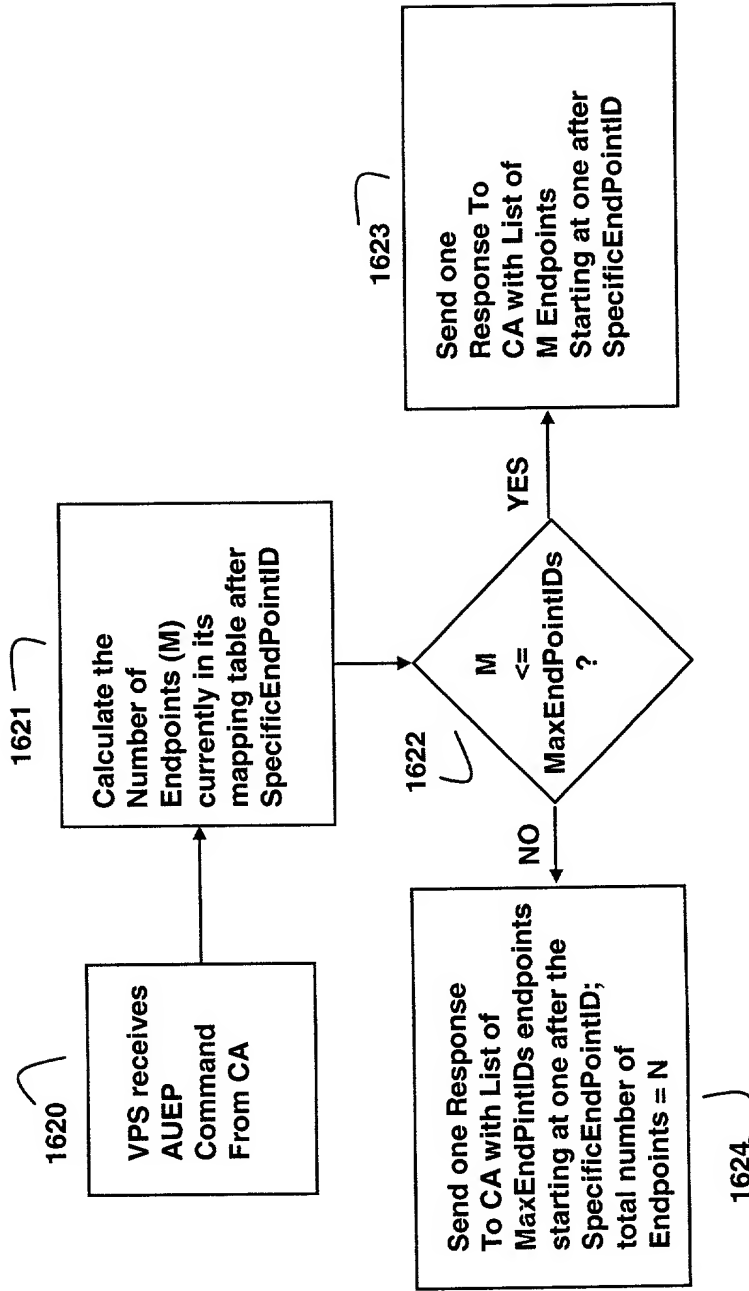


**Audit Multiple Endpoints  
AUEP(EndpointID, MaxEndpointIDs); EndpointID = \*@VPS**



**FIG. 16c**

# **Audit Multiple Endpoints** **AUEP(EndpointID, SpecificEndpointIDs, MaxEndpointIDs); EndpointID = \*@VPS**



**FIG. 16d**

Audit Multiple Endpoints  
AUEP(EndpointID, SpecificEndpointIDs); EndpointID = \*@VPS

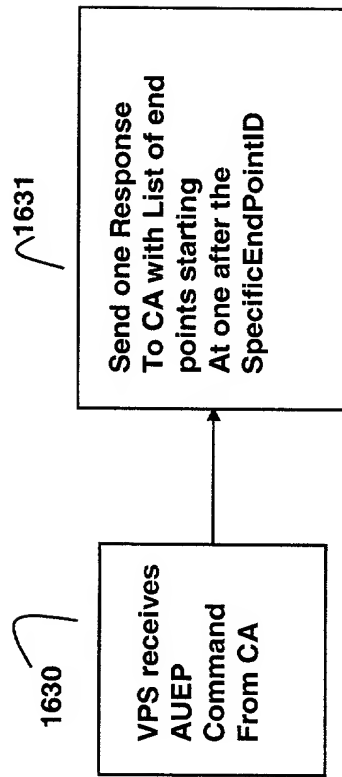


FIG. 16e

# Audit Connection

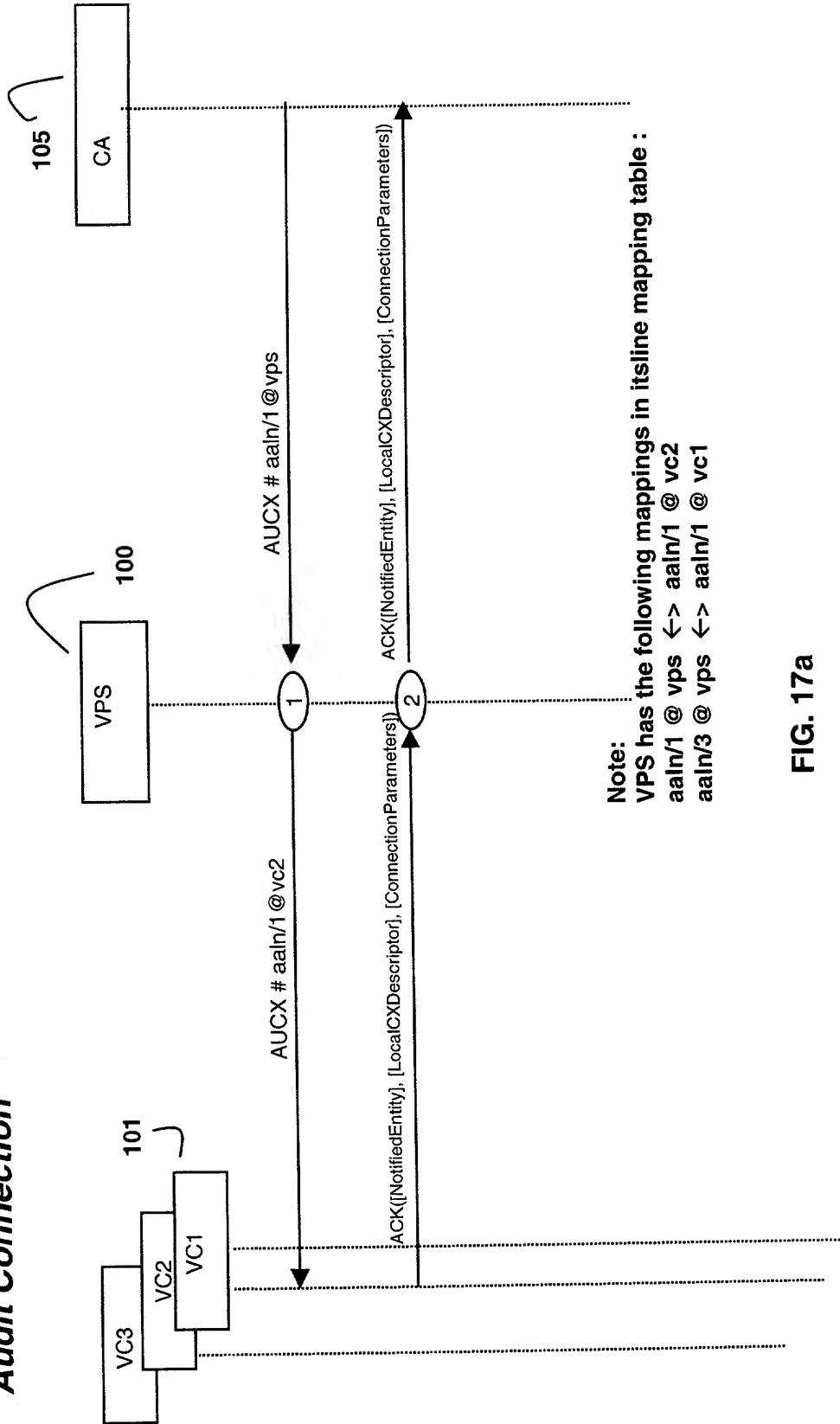


FIG. 17a

Audit Connection  
AUCX(EndpointID, [RequestedInfo]); EndpointID = single endpoint

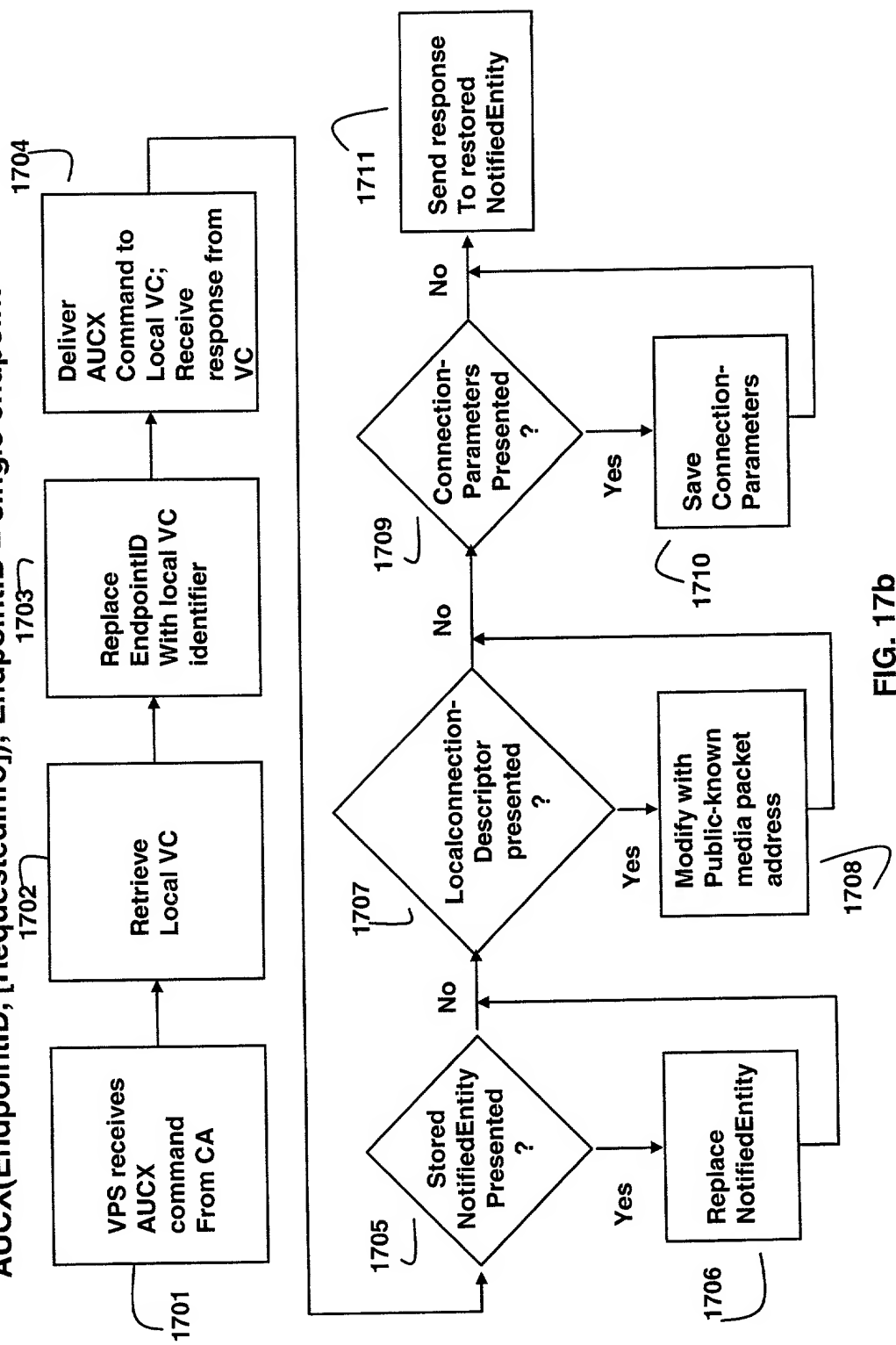


FIG. 17b